FSA Integration Partner

United States Department of Education

Federal Student Aid

Data Strategy Enterprise-Wide Technical Strategies: 123.1.9 Internal Data Strategy

Task Order #123

Version 1.0

July 31, 2003

Executive Summary

The Data Strategy effort relies on the principle that complex interactions between business entities can be broken down into smaller logical functional components, and further understood by assessing the technology enablers along with the business process. This same philosophy is applied to each of the sub-teams that make up the Data Strategy. When topics are divided for analysis it is important that relationships between components be identified and explained.

Several areas of overlap exist between the Internal Data Strategy and other initiatives within both the Technical Strategies and overall Data Strategy efforts. The first such overlap occurs with the remaining four technical areas that will be addressed in separate technical strategy documents:

- Data Storage, Access and Management – The technical components and business processes that define the ability to collect, analyze, access and disburse data.
- External Information Access – The means by which FSA extends Enterprise data and business capabilities to trading partners.
- Web Usage (Portals) – Customer experience and data exchange through the Students Portal, Schools Portal, Financial Partners Portal and other FSA websites.
- Web Services – Software components that use open standard communication protocols to interact with other applications over the Internet for service-orientated architectures.

The Internal Data Strategy also builds upon XML Framework, which outlines the format of data exchanges, a consideration that greatly affects how data can be acted upon and moved between internal systems. Common Identifiers outline the way in which accuracy is obtained between students and external partners, and will leverage the internal data exchange capabilities to ensure valid information is exchanged.

The format, location, and accessibility of data are key components of the overall strategy; they represent an overall Data Approach. The Data Framework effort takes the Data Approach a step further by outlining implementation options to address business issues, and business process changes to allow efficient, accurate exchange of data. This exchange of information leverages the internal data technologies and therefore must be aligned with the Internal Data Strategy. None of the aforementioned areas is wholly independent, each should be treated as logical components that when taken together are building blocks of an enterprise Data Strategy.

The Internal Data Strategy focuses specifically on the way systems within FSA transmit and receive data with one another. The Technical Strategies team collected business input, identified gaps, and with key FSA stakeholders developed options to enable the future vision of internal data exchange. Each of these options were then reviewed and compiled into an overall Internal Data Strategy. Below are the internal data exchange related business objectives and gaps from the Technical Strategies Statement of Strategic Focus. It is worth noting that in the current state, the business objectives as identified by FSA are only minimally realized. Explanations of the assessment criteria can be found in Appendix A: Business Objective Accommodation Criteria.

Rank	Business Objective	Percent Realized	Gaps
1.	Centralized visibility and data flow control of the end-to-end interface process.		• Limited business owner and user visibility exists for transactions that utilize the FSA EAI architecture. • No managed workflow insight into inter-system communications. • Business logic and data transformation not centralized.
2.	Establish common identifiers to enable a reduction of internally exchanged and commonly referenced data.		• Common identifiers for students and schools not utilized across the enterprise. • Common data format does not exist.
3.	Provide Right Time Processing for Internal system communications.		• Legacy systems are not real-time enabled. • Majority of internal data exchange uses batch technologies. • Large, flat files do not readily enable real-time data exchange.
4.	Ability to share customer's status throughout all phases of the lifecycle.		• Data is not available to track customer status. • Architecture not in place to view customer status data. • Clearly defined cross-lifecycle business processes do not exist.
5.	Consolidate Interfaces and provide reuse of services across the enterprise.		• Duplicate data transfers occur between systems. • Manual processes prevent consolidation and reuse of FSA services. • Limited re-use of services exists across the enterprise.

Internal Data Exchange Current State Gap Analysis

This Internal Data Strategy will be combined with strategies from the other four technical areas to represent a single Technical Strategy that is in alignment with the overall FSA Data Strategy effort. The collective picture and the required implementation steps will be outlined in the Technology Vision and Strategic Plan (123.1.12) and the Data Framework Technical Specification (123.1.4). These documents will serve as the enablers of the high-level Data Strategy business objectives.

Overall Recommendation

The overall recommendation for Internal Data Strategy is a target state that is achieved through iterative and deliberate movement to a more real-time, visible and extensible architecture. The solution utilizes a mix of middleware and services-based integration platforms to enable data sharing between FSA systems. Initially, value would be gained through integration of legacy systems via middleware, which would provide transport capabilities as inter-system connectivity evolves to support service-based forms of data exchange.

This mixed approach will also enable the centralization of both inter-system and data transformation logic through the use of process and transformation engines. Where possible, the format of data exchanges should leverage Core Component definitions.

The phased nature of the approach is recommended based upon detailed analysis of business requirements as well as consideration of current industry trends in the area of systems integration. Presently, most data exchange takes place via point-to-point batch transfer. Initially, this will evolve to more real-time centralized means of transport via a hub-and-spoke middleware platform that utilizes a centralized data transformation engine to enable data exchange with legacy systems.

As more FSA systems become real-time enabled, a process engine will be used to enable increased levels of process coordination and transaction visibility across multiple phases of the student aid lifecycle. Once data exchange between systems is mostly real-time and managed by a centralized process engine, it is recommended that FSA migrate to a more services-based approach to integration where groups of like business services are logically grouped and provided for use across the enterprise with Core Component standards used for data format.

Gartner Research supports this move to a process-oriented service-based integration approach: "SOA [Service-Oriented Architecture] is an architectural approach that thrives on turning enterprise computing assets into well-defined services. It is a natural fit with BPM [Business Process Management] because of this reliance on services. SOA exposes services, and BPM, which demands process flow completion, consumes services. If done properly, SOA opens a vast inventory of services for BPM to "glue together into a comprehensive flow."[1]

The business benefits to FSA become clear as the end state is realized:

- Ability to add and remove systems with less effort and impact to other systems.
- Ability to understand transaction flow, history and trending across the enterprise.
- Reduced effort to expose business functions to the external community based on maturity and experience with internal services development.
- Ability to improve data quality by enabling consistent communication methods and formats between systems.
- Ability to access, store, and manage common business logic and capabilities among systems simplifies effort and reduces impact of business process changes.
- The value of legacy systems can be increased through service-based integration.

Next Steps

In preparation for the delivery of the Technology Vision and Strategic Plan (123.1.12), a series of working sessions will be conducted with key FSA stakeholders from July to September as part of the effort to develop strategies for each of the remaining four technology areas. The

[1] © Copyright Gartner, Inc. Source: "SOA and BPM Form a Potent Combination", 14 April 2003.

strategies for each of the five areas will then serve as the basis for the Technology Vision and Strategic Plan, which will be delivered in mid-November.

Amendment History

DATE	SECTION/ PAGE	DESCRIPTION	REQUESTED BY	MADE BY
07/31/03	All	Document submitted for FSA-wide review.	N/A	K. Michl

Table of Contents

Tables and Figures

1 Introduction

1.1 Purpose

The purpose of the Internal Data Strategy is to outline options that may be employed to help FSA reach its internal data exchange vision. In response to FSA Performance Plan Action Item 16.5.3, this document will identify and assess technical enablers that may be used to fulfill business objectives and bridge the gaps for internal data exchange that were identified in the Technical Strategies Statement of Strategic Focus (123.1.6). Strategy documents will be produced for each of the five key Technical Strategies areas in preparation for the delivery of the Technology Vision and Strategic Plan (123.1.12), which will outline the steps required to implement FSA's Data Strategy vision across the enterprise.

1.2 Background

The Department of Education's Federal Student Aid (FSA) organization is seeking to deliver overall improvements in the areas of data quality and data consistency. The goal of implementing improvements to the flow and consistency of data exchange is to ensure that FSA complies with regulations set forth by oversight organizations in support of the program-wide goals that include maintaining a clean audit and ensuring exclusion from the General Accounting Office (GAO) high-risk list.

The effort to improve data exchange between internal FSA systems is one of five technical strategies that when combined with strategies for Consistent Data Framework, XML Framework, Common Identifiers, and Enrollment and Access Management comprise the overall FSA Data Strategy Enterprise-Wide initiative. The other four technical strategies include Web Usage (Portals), Web Services, External Information Access (FSA Gateway), and Data Storage, Management, and Access. Each of these strategies were considered during the development of the of the overall Internal Data Strategy, but will be addressed in separate documents.

In order to create the overall Internal Data Strategy for FSA, the current state of data exchange at FSA was compiled and validated with key business owners. The current state was then used as a basis for a series of seven business objective gathering sessions from which detailed business objectives with respect to the five technical strategies areas were gathered from FSA business owners across the student aid lifecycle.

Approximately 200 raw business objectives were gathered during the seven meetings for each of the technical strategies key areas. These business objectives were then refined, consolidated, and prioritized by FSA business owners in a series of consensus meetings. The business objectives and associated gaps between the objectives and the current state of technology at FSA were gathered and an overall technical strategies strategic vision was presented in the Technical Strategies Statement of Strategic Focus (123.1.6).

The purpose of this document is to detail how improvements in the exchange of data between internal FSA systems can be achieved to help FSA achieve the business objectives that were set forth in the Technical Strategies Statement of Strategic Focus.

1.3 Scope

This document is the first in a series of five deliverables that outline detailed technical options and associated recommendations for each of the five technical strategy areas. The intent of this document is to explain how the business vision presented in the Technical Strategies Statement of Strategic Focus (123.1.6) can be achieved through technology and business process solutions. The strategies associated with each of the five technical areas will later be combined in the Technology Vision and Strategic Plan (123.1.12) to present an overall technical strategy along with a road map for achievement.

The goal of this technical strategy document is to provide a vision for improving the data exchange capabilities between internal FSA systems. The key items that this document will present include:

- An overview of the FSA internal data exchange current state with respect to business objectives.
- Detailed assessment of options to outline the future state of areas that include data transfer, transformation, process logic, and data formatting.
- Analysis performed on each technical option, which includes benefits, drawbacks, and assessment versus a pre-defined set of criteria.
- Recommended technical options that will help FSA best achieve the business objectives that comprise its vision state.

1.4 Assumptions

The following assumptions have been factored into the scope for the Internal Data Strategy:

- A 3-5 year timeframe to implement the technical strategies that will help FSA achieve its business objectives.
- Since the time of the original business objective gathering sessions, additional projects have been identified as having an impact on the Technical Strategies and FSA's future state. As appropriate, these efforts and other potentially relevant efforts will be assessed for impact during subsequent Technical Strategies deliverables.
- Only internal data exchanges are considered. Data exchanges that occur between FSA systems and trading partners are covered by the External Information Access Strategy (123.1.11).
- Considerations have been made that account for the future FSA direction of External Information Access, Web Usage, Web Services, XML Usage, and Data Storage, Management, and Access.
- Real-time data exchange refers to an exchange of data that triggers a series of events across multiple systems as soon as reasonably possible.

1.5 Business Objectives and Gaps

In an effort to begin formulating FSA's technical strategy with respect to internal data exchange, a series of seven meetings were held with business owners from separate phases across the financial aid lifecycle. During these meetings, FSA business owners identified many "raw" business objectives that they felt must be fulfilled in order for FSA to conduct business more effectively.

Once all of the "raw" business objectives were gathered separately for each of the lifecycle phases, a consensus meeting was held with the key business owners from all phases of the student aid lifecycle. During the consensus meeting, the "raw" business objectives were refined, consolidated, and prioritized based on business owner input.

The consolidated business objectives from the consensus meeting were captured in the Technical Strategies Statement of Strategic Focus (123.1.6). These objectives then served as the primary input for the overall vision for internal data exchange presented in the strategic focus document.

The following matrix summarizes the current state of internal data exchange with respect to FSA's consolidated business objectives by identifying the gaps that exist between the current state and the business objectives as they were defined in the Technical Strategies Statement of Strategic Focus (123.1.6). Explanations of the assessment criteria can be found in Appendix A: Business Objective Accommodation Criteria.

Rank	Business Objective	Percent Realized	Gaps
1.	Centralized visibility and data flow control of the end-to-end interface process.		• Limited business owner and user visibility exists for transactions that utilize the FSA EAI architecture. • No managed workflow insight into inter-system communications. • Business logic and data transformation not centralized.
2.	Establish common identifiers to enable a reduction of internally exchanged and commonly referenced data.		• Common identifiers for students and schools not utilized across the enterprise. • Common data format does not exist.
3.	Provide Right Time Processing for Internal system communications.		• Legacy systems are not real-time enabled. • Majority of internal data exchange uses batch technologies. • Large, flat files do not readily enable real-time data exchange.
4.	Ability to share customer's status throughout all phases of the lifecycle.		• Data is not available to track customer status. • Architecture not in place to view customer status data. • Clearly defined cross-lifecycle business processes do not exist.

Rank	Business Objective	Percent Realized	Gaps
5.	Consolidate Interfaces and provide reuse of services across the enterprise.		• Duplicate data transfers occur between systems. • Manual processes prevent consolidation and reuse of FSA services. • Limited re-use of services exists across the enterprise.

Table 1- Internal Data Exchange Current State Gap Analysis

1.6 Key Decision Points

Based upon the business objectives and gaps that were identified in the previous section, a series of key decision points were established and refined by key cross-lifecycle business owners. The purpose of these decision points is to provide guidance in developing options and an ultimate solution recommendation.

Once a set of viable options for internal data exchange was established by the FSA business owners, the following key decision points were used in conjunction with the business objectives and assessment criteria (Section 1.7: Assessment Criteria Methodology) to develop each option:

- **How should batch inter-system data transfer be handled for current and future systems at FSA?**

  The majority of inter-system data exchange currently takes place via a wide range of batch-oriented technologies; however, movement to more real-time exchange may be considered. Continued support for legacy batch transfers may be required.

- **How should real-time inter-system data transfer be handled for current and future systems at FSA?**

  Based on business owner feedback, internal FSA systems should utilize more real-time, methods for inter-system data exchange in order to ensure that each internal system has the "right data at the right time" which is often not the case with the current architecture.

- **What data format(s) should be used in inter-system data exchange at FSA? Should a "common language" be used to facilitate data exchange?**

  Due to the wide range of data formats that are currently exchanged across the enterprise, business owners have recognized the need for standardization to promote data consistency and correctness.

- **How should data transformation be handled and when would it take place?**

  Data transformation services may be required to better enable inter-system data exchange. Consideration should be given as to how data transformation services are provided across the enterprise.

- **How should FSA handle inter-system business logic and process flow?**

  A final aspect recognized by the cross-lifecycle business owners is that little process coordination occurs between systems that serve different phases of the student aid lifecycle. Based upon the current state, methods of uniformly applying business process logic across the enterprise should be considered to better enable process coordination and visibility.

1.7 Assessment Criteria Methodology

In an effort to assess each option effectively, the business owners established a set of assessment criteria. These criteria should be used with the business objectives and key decision points in order to determine each option's true value to FSA.

This set of rating criteria was applied to enable the consistent comparison of options to one another so that it is readily apparent as to which options best serve FSA's future needs. The results of this comparison serve as the basis for determining what the overall technical strategy recommendation is going forward.

The following assessment criteria were used to evaluate each option:

Business Process Impact – Does the option have the potential of introducing significant cross-lifecycle business process improvements? Does the option support the increased ability for real-time data exchange?

Flexibility – Does the option offer improved flexibility for internal data exchange? Can the option support both existing legacy systems and potential future systems?

Level of Effort – Will significant effort (time and resources) be required to realize this option? Will the amount of effort required to maintain internal data exchange be reduced?

User Impact – Will the end user experience improvements with this option?

The following scoring system is used to evaluate each option against the others. For each category, lower numbers represent the less favorable options and higher numbers represent the more favorable options.

Criteria	Rating Scale (1 – 5)
Business Process Impact	1 [Negative Impact] to 5 [Positive Impact]
Level of Effort	1 [Negative Impact] to 5 [Positive Impact]
Flexibility	1 [Negative Impact] to 5 [Positive Impact]
User Impact	1 [Negative Impact] to 5 [Positive Impact]

1.8 Subject Matter Expert Utilization

The following Subject Matter Experts (SMEs) were leveraged during the creation of the Internal Data Strategy:

1. Jonathan Hill, Enterprise Integration Expert – Accenture Global Business Solutions

2 How should data transfer between internal FSA systems be handled?

As was identified in the Technical Strategies Statement of Strategic Focus (123.1.6), many of the internal data exchanges at FSA are batch transfers that take place via custom point-to-point interfaces and utilize a variety of proprietary, system-specific flat file formats. Some FSA internal data exchange takes place in real-time via the FSA Enterprise Application Integration (EAI) Architecture and in the case of a few transactions utilizes advanced file formats such as XML. A very small number of transfers take place via manual means such as magnetic tape or spreadsheet.

Given the wide range of data transfer methods that are used to transfer data between FSA systems, a consistent, centralized approach to internal enterprise data transfer is required to ensure data quality while facilitating efficient data exchange. Based upon the business objectives previously identified, three options for enabling inter-system data transfer should be considered:

- Middleware Based Integration
- Service Based Integration
- Customized Integration

Based on feedback gathered in the business objective meetings and working sessions with key FSA business owners, the architecture options considered to enable internal data exchange should meet the following business requirements:

- Large, batch-style data transfers with latency must be supported for the near future.
- The future direction of inter-system data exchange at FSA must include increased levels of real-time data exchange.
- Business logic that is currently duplicated across multiple internal FSA systems should be provided as centrally accessible services within the FSA enterprise.
- The architecture must be capable of supporting increased numbers of small, real-time synchronous transactions.
- The architecture must be capable of supporting increased numbers of small, real-time asynchronous transactions.

2.1 Middleware-based Integration

One possibility for enabling data exchange between internal systems is to utilize a middleware platform to handle both batch and real-time transactions. This option would satisfy a near-term business requirement by providing support for existing batch transfers. The middleware platform can be used in either a point-to-point or hub and spoke configuration depending on the application integration requirements.

In this option, the centralized transport layer of a middleware platform is utilized to transport both batch and real-time data between internal FSA systems. Once a centralized message transport layer is in place, additional capabilities such as data transformation and process logic

can be easily applied to the overall integration solution. Options for these capabilities will be discussed in Section 4.

In the point-to-point configuration, connectivity is enabled between two distinct systems with the assumption that the data being exchanged is only needed by the two systems involved in the exchange. Each system is connected to the middleware platform via a set of application connectors that are used to place data onto and pull data off the integration platform.

The hub and spoke middleware configuration is ideal for enabling systems to exchange data with any number of destination systems. As in the point-to-point configuration, each system has a set of application connectors that enable connectivity to the integration platform; however, data is then routed by the middleware platform as necessary to multiple systems.

The middleware-based integration solution will meet the immediate needs for internal systems because of the increased levels of flexibility that the platform provides. This flexibility allows a phased approach to be taken thus allowing systems to be integrated in logical groupings in reasonable timeframes that would minimize impacts to ongoing FSA business. In addition, the middleware platform's flexibility will allow for the relatively seamless inclusion of future systems into the overall integration architecture.

The primary benefit of using a middleware architecture for inter-system data exchange is its centralized message transport capabilities that are capable of providing guaranteed levels of message delivery and standardized error handling. Benefits could also be gained through the inclusion of transformation and business logic facilities. In this area, the hub and spoke middleware configuration provides benefits over the point-to-point configuration in that transformation and business logic can be centrally located and applied to multiple data exchange points thus reducing the overall complexity of the architecture.

The middleware architecture is also capable of handling multiple data formats due to its ability to perform data transformation and mapping to multiple systems. When it makes business sense movement to XML with Core Component definitions should occur. The overall data format aspect of the Internal Data Strategy is discussed in Section 4.

This option presents the following benefits to FSA:

- Provides a standardized approach for integrating systems across the enterprise.
- Supports both batch and real-time data transfer.
- Provides a means for integrating legacy applications.
- Allows for a phased approach to implementing new architecture with the intent of minimizing impact to FSA's overall business.
- Supports the application of data transformation and process logic components.
- Provides a starting point for centralized tracking and monitoring of transactions across the enterprise.

The following diagram illustrates the potential use of middleware-based integration:

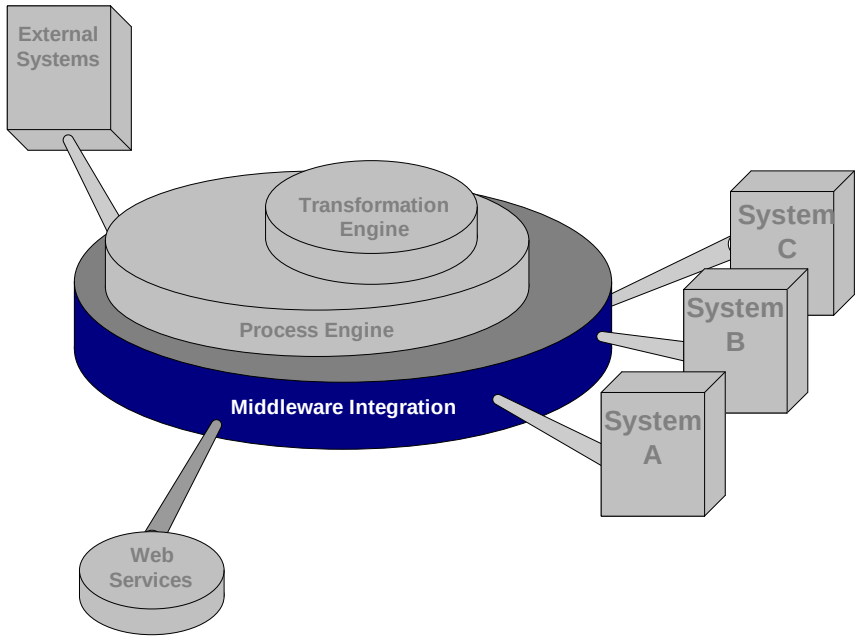

Figure 1 - Middleware-based Integration

2.1.1 Option Assessment

The following tables present assessments of middleware-based integration as it applies to business process impact, flexibility, level of effort, and user impact:

Business Process Impact	Does the option have the potential of introducing significant cross-lifecycle business process improvements? Does the option support the increased ability for real-time data exchange?	1……2……3……4……5 Negative Business Process Impact — Positive Business Process Impact

Points	**Comments**
Provides accurate message flow and status so that business owners know the status of transactions.	• Integration platform will help enable the integrated customer view by providing data transport capabilities. • Integration platform access to enterprise data stores will enable sharing of historical customer information across the enterprise.
Reduced need for transaction reconciliation.	• The middleware platform provides guaranteed levels of message delivery and persistence thus allowing for less manual reconciliation.
Business process changes may be required.	• Current business processes may be optimized for custom point-to-point integration.

Table 2 - Middleware-based Integration Business Process Impact Assessment

Flexibility	Does the option offer improved flexibility for internal data exchange? Can the option support both existing legacy systems and potential future systems?	1…..……2…..…..3……..…4.……5 Not Flexible — Highly Flexible

Points	Comments
Increased data exchange flexibility.	• A middleware will allow for relatively seamless sharing of data among many internal FSA systems. • Legacy systems can be integrated using native file formats where necessary to minimize impact to system owners.
A middleware-based architecture supports phased implementation and rollout.	• Source systems can be connected to the integration platform at cost and resource effective rate. • Systems can be added to the integration solution in logical groupings (e.g., similar business functions, etc.).
Improved capabilities to add new systems to the enterprise.	• A centralized integration approach will allow the improved ability to integrate new application systems into the enterprise technical architecture as required. For example, if a legacy system is being replaced there is no need for systems that interact with the legacy system to know about it as they communicate only through the integration platform.

Table 3 - Middleware-based Integration Flexibility Assessment

Level of Effort	Will significant effort (time and resources) be required to realize this option? Will the amount of effort required to maintain internal data exchange be reduced?	1…..……2…..…..3……..…4.……5 Significant Effort Required — Minimal Effort Required

Points	Comments
Minimal effort is required to integrate systems that currently use the EAI architecture.	• Systems that are currently EAI enabled will require relatively little effort to integrate.
Maintenance and support efforts would likely be simplified and reduced.	• Utilizing a single centralized means for integration will decrease the overall maintenance and support complexity because middleware inherently reduces the number of potential failure points.
Minimal effort is potentially required to integrate new systems.	• New systems can easily be integrated with systems currently connected via the middleware platform.
Effort may be required to enable legacy system integration.	• Some amount of custom data transformation logic may be required for legacy systems. • Many internal FSA systems do not currently support real-time data exchange.

Table 4 - Middleware-based Integration Level of Effort Assessment

User Impact	Will the end user experience improvements with this option?	1…..……2…..…..3……..…4.……..5 Negative Impact to User — Positive Impact to User

Points	Comments
User roles may be simplified.	• Workflow and business process automation can simplify the data exchange roles of FSA end users. • Total end-to-end visibility into the data exchange process across FSA will result in less manual reconciliation.
Adaptation to new business processes may be required.	• Users will likely have to accept and adapt to new business processes that support middleware-based integration.

Table 5 - Middleware-based Integration User Impact Assessment

Score Summary

Business Process Impact	Flexibility	Level of Effort	User Impact
4	5	4	3

Table 6 - Middleware-based Integration Assessment Scoring

2.2 *Service-Based Integration*

Another option to enable FSA's Internal Data Strategy is to leverage the capabilities of a service-based architecture. In a service-based architecture, discrete business services required by multiple systems are enabled in a single, centrally accessible manner. The Effective Family Contribution (EFC) Calculator is an example of a business service that could be enabled through a service-based architecture.

This emerging technical architecture pattern follows current technology trends by effectively abstracting business process functions from specific systems and locations. The specific business functions are then usually exposed in the form of centrally located Web services. This form of integration will support a phased implementation approach because logical groupings of business process functions could be introduced at differing times based upon business need.

Unlike the previous option for internal data exchange, which potentially requires system knowledge to enable data exchange between multiple systems, this architecture option operates under the principal that a uniformly accessible capability is enabled and multiple systems are then allowed to access it. For example, if multiple systems require customer addresses to be periodically updated, an "Update Address" service could be created to uniformly process address updates across all systems. In addition to new address data, the "Update Address" service might take arguments indicating the name of the source and destination systems, thus allowing any system to update addresses on other systems without having knowledge of the address implementation on the destination system.

Integration trends support this integration approach in that the current direction is to construct new systems by employing service-oriented architecture internally leveraging the loosely coupled characteristics of services.

Typically, application connectivity to services is handled by transmitting a Simple Object Access Protocol (SOAP) request via standard Internet transport protocols; however, considerations must be made concerning the volume and frequency of data received in order to determine how the SOAP request will be delivered. A full examination of Web services transport protocols will be presented in the Web Services Strategy (123.1.8) document.

In this case, application connectivity to services will be handled by either standard transport protocols or the transport layer of a middleware platform. The reason for the combination is that Internet transport protocols are not well suited for transferring large amounts of data (as in the case of batch transfer). The transport layer of the middleware platform is better suited than Internet protocols for large bulk transfers. The middleware transport layer should be leveraged to deliver SOAP requests because:

- Middleware platform provides the ability to transfer large volumes of data.
- Existing infrastructure can be leveraged to provide guaranteed levels of delivery.
- Enables the deployment of asynchronous services, whereas Internet protocols typically require styles of exchange that are more conversational.

Industry trends support the use of middleware as the transport layer in service-based exchange because of the high-levels of guaranteed throughput that the platform delivers.

Web services are typically enablers of service-based architectures, and consequently the data format required to exchange data is XML. When XML is used, emphasis should be placed on mapping to Core Component definitions whenever possible. The overall data format aspect of the Internal Data Strategy is discussed in Section 4.

While being slightly more advanced than the previous architecture presented, this option offers the following distinct capabilities:

- Ability for services to mirror actual business processes as opposed to discrete pieces of application functionality.
- Ability to utilize system services without in-depth knowledge of destination systems and data exchange formats.
- Simplified discovery of service and return value definitions without communication between systems.
- Increased support for real-time request/response transactions.

The service-based integration approach should be considered as a viable option for enabling connectivity to future batch and real-time based FSA systems with the intent being that all internal data transfers eventually be service-based.

This option presents the following benefits to FSA:

- Provides for centralization of key business services across the enterprise.
- Enables evolution to a more real-time data exchange environment.
- Readily supports real-time data exchange.
- Minimizes business impact to FSA through a phased implementation approach.
- Provides the framework for enablement of centralized process and data transformation logic.
- Readily enables the centralization of transaction tracing and centralized monitoring across the lifecycle.

The following diagram represents a possible use of a service-based architecture:

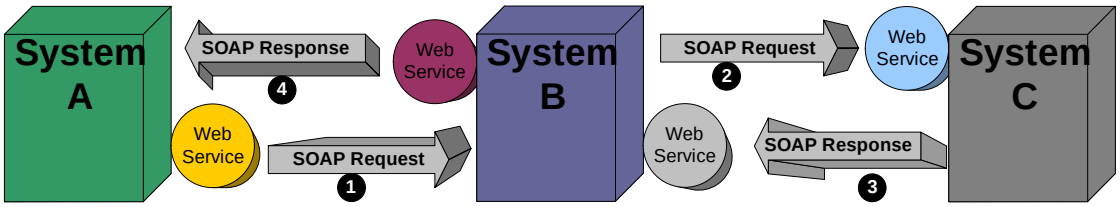

Figure 2 - Service-based Integration

2.2.1 Option Assessment

The following tables present assessments of service-based integration with respect to business process impact, flexibility, level of effort, and user impact:

Business Process Impact	Does the option have the potential of introducing significant cross-lifecycle business process improvements? Does the option support the increased ability for real-time data exchange?	1……2……3……4……5 Negative Business Process Impact — Positive Business Process Impact

Points	**Comments**
Potential for customer service improvement.	• The centralization of business functions would promote consistency by allowing for the uniform application of services across FSA systems.
A new set of business processes will need to be created to support service-based integration.	• Existing business processes support batch and EAI-based data transfer methods only. • Current business processes do not readily support the sharing of services across the enterprise – service owners would need to be defined.

Table 7 - Service-based Integration Business Process Impact Assessment

Flexibility	Does the option offer improved flexibility for internal data exchange? Can the option support both existing legacy systems and potential future systems?	1…….2…….3…….4…….5 Not Flexible — Highly Flexible

Points	Comments
Simplified service distribution.	• Service-based integration will allow for easy and uniform distribution of services across multiple systems.
Option supports both batch and real-time data exchange.	• Internet protocols readily support small, lightweight data exchanges. • Middleware transport layer supports large, bulk-style transfers.
Option supports a phased implementation.	• Services can be activated and integration rolled out at a cost and resource effective rate. • Services can be added to the integration solution in logical groupings (e.g., similar business functions, etc.).
New systems are easily supported in the integration solution.	• A service-based integration approach will allow the improved ability to add new application systems into the enterprise technical architecture as required.

Table 8 - Service-based Integration Flexibility Assessment

Level of Effort	Will significant effort (time and resources) be required to realize this option? Will the amount of effort required to maintain internal data exchange be reduced?	1…….2…….3…….4…….5 Significant Effort Required — Minimal Effort Required

Points	Comments
Reduced effort to access and use cross-system services.	• Services are discovered and accessed dynamically, thus reducing the requirement to know system functionality intimately. • Services can be used and re-used by multiple systems across the enterprise.
Level of effort required to integrate new systems is relatively small.	• Core component integration promotes a common integration language between systems.
Proven standards can be leveraged.	• Service-based architectures are typically based on Web services which employ standards such as XML, SOAP, etc.
Legacy systems can be included in service-based exchange.	• Current technology trends allow for the inclusion of legacy systems in service-based architectures. • Legacy application functionality would need to be "wrapped" for inclusion in service-based integration. • Wrapping legacy system functionality is a short-term solution for providing access to legacy system services.

Table 9 - Service-based Integration Level of Effort Assessment

User Impact	Will the end user experience improvements with this option?	1…….2…..…..3……..4.……5 Negative Impact to User — Positive Impact to User

Points	Comments
Uniform access to and utilization of central services.	• Users can easily locate and access enterprise-level services. • Services are provided consistently across enterprise.
Potential for simplified user roles.	• Total end-to-end visibility into the data exchange process across FSA will result in less manual reconciliation.
Users required to adopt a new set of business processes.	• Users will likely have to accept and adapt to new business processes that support service-based architecture.

Table 10 - Service-based Integration User Impact Assessment

Score Summary

Business Process Impact	Flexibility	Level of Effort	User Impact
4	5	4	4

Table 11 - Service-based Integration Assessment Scoring

2.3 Customized Integration

A final option for enabling connectivity between internal FSA systems is to leverage custom integration platforms. The term Customized Integration refers to any form of system-to-system connectivity that takes place via non-Commercial off the Shelf (COTS) platforms. Custom integration solutions typically provide for direct connectivity between two distinct systems and provide for little, if any, data sharing with other systems within the enterprise. Current examples of custom integration include direct database-to-database connectivity and FTP-based point-to-point transfers between systems.

Customized integration remains a possible option for enabling internal inter-system connectivity at FSA. However, the use of customized integration solutions should be minimized as to not cause adverse impacts on centralization and standardization that are key requirements set forth by FSA business owners. Potential scenarios when custom integration might be considered as a data exchange alternative include:

- Providing connectivity between two distinct legacy systems where high throughput levels are required.
- High performance requirements exist that cannot be met by a centralized integration approach.
- Geography preventing connectivity between a particular system and other systems through a central integration platform.

2.3.1 Option Assessment

During a set of working sessions, FSA business owners recognized that custom integration solutions would not readily meet their future needs for the following reasons:

- Custom solutions present in the current architecture do not promote high-levels of reuse.
- Complexity of current custom integration solutions limits FSA's ability to manage integration across different areas of the enterprise and student aid lifecycle.
- Custom integration solutions tend to promote tight coupling between systems.

The business owners did recognize, however, that each existing custom integration solution would need to be carefully analyzed as part of the detailed design process to ensure that all needed functionality is included and that the full set of business requirements is met.

3 How should business logic be performed and increased transaction visibility gained?

One area that the cross-lifecycle business owners identified for improvement is the application of inter-system business logic and the capability to track data effectively as it moves throughout the enterprise. Many business owners have a strong desire for automating business processes that cross multiple systems and lie in different phases of the student aid lifecycle. In addition, there is a distinct need for the ability to track data across all systems and phases of the lifecycle. The following sections address both data transformation and the application of business process logic.

3.1 Data Transformation

In each of the integration options presented in the previous section, certain levels of data transformation logic will be required to enable effective data exchange between systems. Presently, a high number of customized data transformations take place in internal data exchange, and many of these transformations are duplicated. The high number of transformations and duplication is due in part to the increased number of custom point-to-point integration solutions that exist between FSA systems.

As FSA moves toward a more centralized approach to internal data exchange, a uniform approach to the application of data transformation logic must be adopted across the enterprise. Additionally, consistent data format and content standards must be applied to data that is shared between systems. For each of the data transfer options presented, data transformation logic can be utilized from various platforms and applied in either decentralized or centralized manners depending upon business requirements that include:

- Increased amounts of business transformation logic will be required as the volume of real-time data exchange increases as part of FSA's overall internal data exchange vision.
- Consideration must be given to the number of systems with which distinct pieces of data is being exchanged.
- Performance and visibility requirements need to be fully understood to ensure that the detailed design is based upon accurate requirements.
- Redundant data transformations between internal FSA systems should be reduced.
- Data transformation to Core Component definitions and XML format should occur when data is to be exchanged with two or more systems.

Independent of the integration approach that is taken, data transformation logic should be centralized with respect to the integration platform when it will be heavily used by multiple systems across the enterprise. While some transformation logic may be applied at the application connector level, the application of transformation logic in this manner should be minimized to reduce troubleshooting complexity as well as the overall number of connector instances that are required.

One approach to the application of data transformation logic is to employ it in a decentralized manner across all enterprise systems. In this configuration, individual systems across the enterprise independently own and manage their own transformation logic. A decentralized approach should be considered in the following scenarios:

- Data is being exchanged between no more than two systems with little potential for future exchange with other systems.
- Performance requirements do not allow for the centralization of data transformation logic.

The benefits of a decentralized approach to applying transformation logic include:

- Potential for increased inter-system data exchange performance due to the distributed application of transformation logic.
- Transformation logic is developed and maintained by each system owner; therefore, it can be assumed to be correctly and accurately applied.

Another possible approach is to apply data transformation logic in a centralized location as part of the integration platform or as services provided to the integration platform itself.

A centralized approach for data transformation should be considered in the following situations:

- When data is being exchanged with multiple systems across the enterprise that require varying data formats to facilitate exchange.
- If data exchange is occurring between only two systems with potential for future exchange with other new or existing systems.

The benefits of a centralized approach to applying transformation logic include:

- Reducing the number of locations where data transformation takes place.
- Providing a consistent approach for data transformation that can be applied across the enterprise.

In the near term, logic will be required to enable legacy file transformations to common definitions. However, over time this need will be greatly reduced as many systems migrate to Core Component content and data standards with the ultimate responsibility placed upon each distinct application system to communicate with others using Core Components.

The following diagram illustrates transformation of system-specific data to Core Component standards. It should be noted that only data that will be exchanged between multiple systems would be transformed – all non-shared system specific data will remain in legacy system formats.

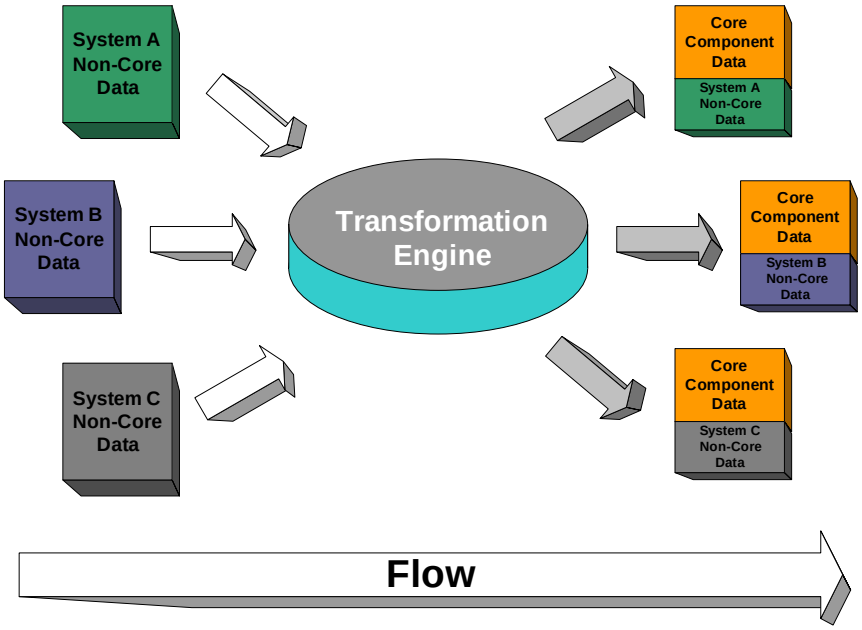

Figure 3 - Data Transformation Example

3.2 Process Management and Transaction Visibility

Regardless of the integration approach taken, a requirement has been identified that centralized process management and increased transaction visibility be integrated into the cross-lifecycle data exchange solution. Currently, there is limited coordination of business processes across the financial aid lifecycle and very little visibility into the various inter-system data exchanges that take place.

As part of the movement toward more centralized forms of internal data exchange, the ability to perform business process coordination and trace data as it moves throughout the enterprise are key enablers for the FSA internal data exchange vision. To enable these abilities, inter-system business logic should be applied through a centralized process engine based upon the following business requirements:

- Increased visibility into the end-to-end, cross-lifecycle data flow.
- Centralized monitoring of each system's transaction interface points across the enterprise.
- Process management across multiple phases of the lifecycle.
- Ability to apply monitoring and process logic in an iterative approach.

The centralized process engine would provide the ability to apply inter-system business logic uniformly across aspects of the enterprise and all phases of the student aid lifecycle. The process engine would also enable the capability for detailed data tracing and routing through virtually every system touchpoint.

The initial approach for applying business process logic is to apply it to processes where business benefits will be realized. Different groups of systems would be enabled on the process engine platform in logical groupings as to minimize the overall impact to FSA's ongoing business.

The benefits of providing centralized business process logic include:

- Centralized process engine provides the ability to automate fully data transfer between systems across different phases of the student aid lifecycle.
- Leveraging a business process engine will encourage the implementation of similar business processes to adopt a common architecture approach, thus maximizing the use of other architectural aspects such as data transformation and transport.
- Utilizing a process engine will allow business owners to quickly understand and modify process steps and models.
- Enabling error handling and process management will improve the end user experience through quicker remediation of data transfer issues.
- Increased ability to perform meaningful and quick data reconciliation.
- Process logic can be extended to incorporate data exchange that occurs with external entities as well to enable collaboration that is more robust.

The following figure illustrates the use of a centralized process engine to manage state and route data between internal systems:

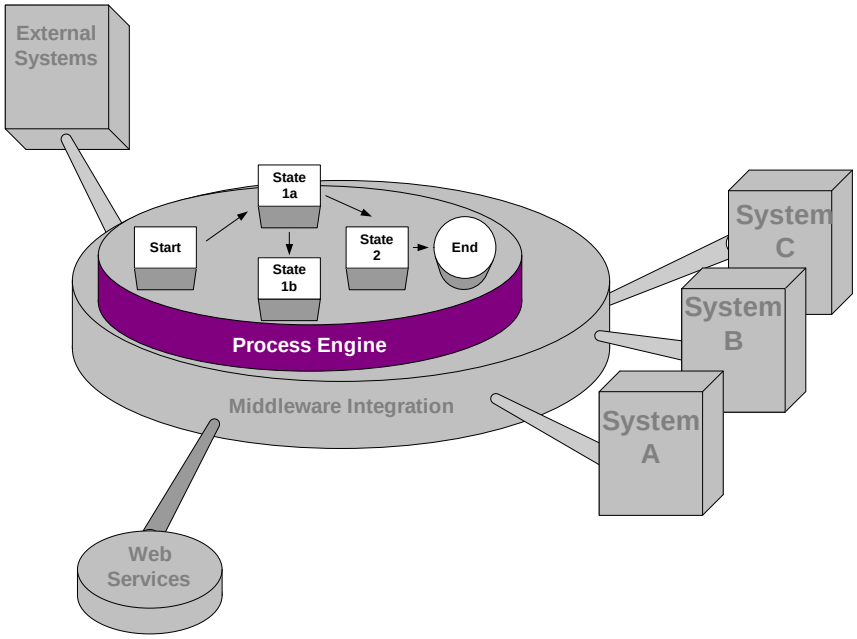

Figure 4 - Process Engine Example

3.3 Option Assessment

The following tables present assessments of how business logic will be applied to the overall integration solution with respect to business process impact, flexibility, level of effort, and user impact:

Business Process Impact	Does the option have the potential of introducing significant cross-lifecycle business process improvements? Does the option support the increased ability for real-time data exchange?	1…..……2…..…..3…….…4.……5 Negative Business Process Impact — Positive Business Process Impact

Points	**Comments**
Phased approach to applying business process logic minimizes negative impact to FSA.	• Business process logic can be applied to groups of systems in timeframes that make business sense for FSA.
Cross-lifecycle business processes can be automated.	• Process engine and business logic services support the complete automation of end-to-end business processes across the enterprise.
Improvements to data tracking and reconciliation processes.	• When in place with an assured delivery framework such as a middleware platform, quick access is available for transfer status, failures, etc. • Enables tracking of student status throughout the financial aid lifecycle.
Business process modifications may be required to enable the cross-lifecycle application of logic.	• Little process coordination exists across different phases of the student aid lifecycle. • Business owners may have to modify their internal processes to enable cross-phase process application.

Table 12 - Transformation and Process Logic Business Process Impact Assessment

Flexibility	Does the option offer improved flexibility for internal data exchange? Can the option support both existing legacy systems and potential future systems?	1…..……2…..…..3…….…4.……5 Not Flexible — Highly Flexible

Points	**Comments**
Addition of new systems to the process management solution is relatively easy.	• The process engine is the central location for most inter-system business logic. • New processes can be integrated easily into the solution, as the need exists.
Standardization of business processes, and integration services.	• Promote the development of a common architecture approach. • Potentially maximizes the reuse of components across the enterprise integration solution.

Table 13 - Transformation and Process Logic Flexibility Assessment

Level of Effort	Will significant effort (time and resources) be required to realize this option? Will the amount of effort required to maintain internal data exchange be reduced?	1…..……2…..…..3……..…4.……..5 Significant Effort Required — Minimal Effort Required

Points	Comments
Reduced effort will be required to manage and update business processes going forward.	• Central process management allows for relatively easy process additions and modifications.
Effort may be required to implement and manage business processes across the student aid lifecycle.	• Input will be required from each of the business owners to correctly outline and implement cross-lifecycle business process logic. • Effort and coordination will be required between business owners and individuals implementing the process solution to ensure that the process is kept up-to-date.

Table 14 - Transformation and Process Logic Level of Effort Assessment

User Impact	Will the end user experience improvements with this option?	1…..……2…..…..3……..…4.……..5 Negative Impact to User — Positive Impact to User

Points	Comments
Reduced effort to resolve transfer issues.	• Improved error handling and process management will lead to faster issue resolution.
Centralized transformation promotes improved data exchange with external partners as well as internal systems.	• External data exchange points can leverage transformation logic to enable data exchange with the internal FSA environment. • Potential re-use of transformation logic results in minimized data format changes that will be required for data that originates from external systems.

Table 15 - Transformation and Process Logic User Impact Assessment

Score Summary

Business Process Impact	Flexibility	Level of Effort	User Impact
4	5	3	4

Table 16 - Transformation and Process Logic Assessment Scoring

4 What data considerations must be made?

As outlined in the current state analysis in the Technical Strategies Statement of Strategic Focus (123.1.6), many different data formats are used to facilitate data exchange between internal FSA systems with few standards being followed with respect to structure and content. The FSA business owners have recognized the need to establish and follow data format and content definitions. Standard data format and definitions will provide a consistent approach to data exchange and assist in promoting increased data quality levels across the FSA enterprise.

The following sections discuss the application of format and content standards to inter-system data exchange at FSA. It is also worth noting that these standards also apply to data exchange with external partner, which will be covered in the External Information Access (FSA Gateway) Strategy (123.1.11).

4.1 Core Components Background

As presented in the XML Framework Strategic Assessment and Enterprise Vision (123.1.13), the Core Components effort originated with the Common Origination and Disbursement (COD) Common Record. The COD Common Record utilizes data content as well as XML formatting standards to enable uniform data exchange between FSA and school trading partners.

Following on the success of the COD Common Record, efforts are currently underway to draft the Central Processing System (CPS) Institutional Student Information Record (ISIR) as an XML schema and additional initiatives are planned both within FSA and the Postsecondary Financial Aid Community.

The purpose of the Core Components initiative is to describe commonly referenced data as a set of XML schema entities. These will serve as building blocks for data exchanged across the student aid community. The primary goal of the Core Components initiative is to improve the overall data quality at FSA through standardization, and the resulting minimization of ambiguity that currently exists in data definitions for system interfaces across the enterprise.

4.2 Application of Core Components and Standardized Formatting

The concept of Core Components is a key aspect of the implementation of standardized data formats for use in internal data exchange across FSA, and is a concept in the overall FSA data cleanup effort. Core Components will enable the comparison of data between systems through the application of internal data standards for both content and formatting. These internal data standards will provide a single set of modeled components that will help to eliminate data ambiguities in and facilitate inter-system data exchange. Ultimately, the establishment of Core Components standards will serve to further facilitate and improve the quality of internally exchanged data. It should be noted that Core Component definitions will not be created for all data elements exchanged within FSA only for the data that is shared across multiple systems.

For data exchange that utilizes the XML file format, a centralized, standards-compliant Registry and Repository will be created to publish and house Core Component definitions. These will be used to define the structure and data content requirements for each Core Component. Core Components will be leveraged to form complete messages. A complete message may contain both Core Component data and interface-specific data.

For systems where the use of XML file format is not possible due to technological limitations, fixed-width flat files will continue to be used for data representation. Presently, many legacy systems use different proprietary fixed-width file formats for data exchange, which adds complexity in terms of transformation and redundancy when attempting to enable data exchange with other systems within the enterprise. In order to simplify flat file data exchange, the concepts that are presented in the Core Component definitions can be applied to flat file formats to further assist in data quality improvement. The centralized Core Components Registry and Repository will be used to publish definitions and standards to interested parties across the enterprise.

4.3 Core Components Standards with Respect to Data Exchange and Transformation

As previously discussed, the FSA business owners have identified the ability to perform centralized data exchange and transformation as key business requirements going forward. The application of Core Component definitions and standardized data formatting plays a key role in FSA's ability to achieve each of these goals.

The main goal is twofold: standardize the definition of data elements, and standardize the format and representation of the data elements. The application of Core Component and data formatting standards will evolve over time with the internal data exchange architecture, as few systems within FSA are capable of generating data based upon Core Component format and content standards. Initially, the internal data exchange architecture will need to have the capability of mapping commonly exchanged elements into Core Component definitions to allow for better exchange with multiple systems.

As the internal data exchange architecture evolves to a more real-time, service-based architecture, it is likely that interfaces for more FSA systems will be defined based upon standards that exist in the Core Components Registry and Repository. These interfaces will have the capability to generate XML-formatted data based upon Core Component definitions and format standards. Overall, this standardization will result in a reduction in the amount of data transformation that will be required in internal data exchange.

4.4 Option Assessment

The following tables present assessments of how Core Component definitions will be applied to the overall integration solution with respect to business process impact, flexibility, level of effort, and user impact:

Business Process Impact	Does the option have the potential of introducing significant cross-lifecycle business process improvements? Does the option support the increased ability for real-time data exchange?	1……2……3……4……5 Negative Business Process Impact — Positive Business Process Impact

Points	Comments
Business owners can adopt Core Component definitions and file formats over time.	• Many FSA systems cannot be easily converted to use XML-based file formats for data exchange. • The middleware-based integration platform will allow for a phased application of Core Component definitions to legacy systems via centralized data transformation capabilities.
Application of Core Components reduces data exchange complexity.	• Data definitions are currently not in place to aid in interface definitions. • Defining future interfaces will be simplified through the application of Core Component definitions.
Process changes may be required to utilize Core Component Standards.	• Data is currently defined in many different ways across multiple FSA systems.

Table 17 - Data Format Business Process Impact Assessment

Flexibility	Does the option offer improved flexibility for internal data exchange? Can the option support both existing legacy systems and potential future systems?	1……2……3……4……5 Not Flexible — Highly Flexible

Points	Comments
The application of Core Components supports high-levels of data reuse across the enterprise.	• Data definitions are standardized across the enterprise. • Systems making use of Core Component definitions when exchanging data will likely require minimal amounts of data transformation logic.

Table 18 - Data Format Flexibility Assessment

Level of Effort	Will significant effort (time and resources) be required to realize this option? Will the amount of effort required to maintain internal data exchange be reduced?	1……2……3……4……5 Significant Effort Required — Minimal Effort Required

Points	Comments
Core Component definitions will allow for the easier integration of new internal systems.	• Standard data definitions will allow for easier modeling of new interfaces for use in enterprise systems integration.
Core Component definitions will help FSA realized long-term maintenance savings.	• Up-front effort will be required to learn and adhere to standards. • As standards are uniformly accepted, maintenance effort associated with data exchange will be reduced.
Effort may be required to enable systems to exchange data based upon Core Component definitions.	• Custom transformation logic may be necessary to map legacy data into Core Component definitions. • Data may be defined in multiple ways across many systems within the enterprise.

Table 19 - Data Format Level of Effort Assessment

User Impact	Will the end user experience improvements with this option?	1….……2…..…..3……..…4.……5 Negative Impact to User — Positive Impact to User

Points	Comments
Effort required to reconcile data exchanges will be reduced.	• Data reconciliation processes may be created based upon Core Component definitions. • Standardized data definitions and formats reduce data exchange complexity; however, system specific information must still be defined. • Enterprise data definitions and content standards are clearly provided and readily available to users.

Table 20 - Data Format User Impact Assessment

Score Summary

Business Process Impact	Level of Effort	Flexibility	User Impact
4	5	3	4

Table 21 - Data Format Assessment Scoring

5 Recommendation

Creation of an Internal Data recommendation requires consideration of FSA business needs, industry trends, and understanding of potential return on investment. The rational for change must be in improved operational efficiencies or capabilities of an enterprise. This recommendation is only part of the overall picture – the Data Strategy is unique in that it takes into consideration FSA cross-lifecycle business processes and requires a thorough understanding of the data exchanges between both internal and external systems/partners. This "big think" approach helps to ensure that a solution option has acceptable trade-offs across all of FSA. It should be noted that no single solution fits all, and furthermore the implementation of a new strategy and architecture requires a carefully planned and iterative approach.

The Internal Data Strategy is the first of five individual strategies from the Technical Strategies team. The recommendations are made with respect to the larger picture of Data Strategy across all five of the Technical Strategies, however each piece must be broken down to understand the value of an end solution. The final deliverable produced by the Technical Strategies team will highlight the complete Technical Strategy vision and will "assemble" the five individual strategy components. Alignment with the Data Strategy vision is necessary, and the technologies should enable the business while bringing new capabilities and efficiencies to fruition.

The recommendation below builds off the analysis of technical options for each area and their ability to help FSA achieve its overall business goals, fill outlined gaps and integrate with an overall data vision. Internal data is a key piece of the strategy as it enables system-to-system communication and has a large impact of the day-to-day operations at FSA. The following components comprise the overall strategy for internal data exchange at FSA.

- Inter-System Data Exchange
- Data Transformation
- Business Process Logic
- Data Format

Each of the above pieces helps to mold an overall internal data capability. Additionally, this internal capability furthers the enterprises ability to exchange, store, and manage data both internally and externally.

5.1 Recommendation Summary

A phased approach with elements of both hub-and-spoke middleware and service-based elements is recommended to enable data exchange between internal FSA systems. Initially, batch and real-time data exchange should be enabled though hub-and spoke middleware, with movement towards a more service-based state as the enterprise evolves toward increased levels of real-time exchange. The middleware-based approach provides immediate value through the integration of legacy systems and over time will serve as the basis for service transport.

Gartner Research supports the move to a service-based architecture for real-time data exchange: "Service-oriented architecture is a best-practice architecture pattern for the systematic design of request/reply applications. Its primary intentions are business-level software modularity and rapid, nonintrusive reuse of business software in new runtime contexts."[2]

The mixed approach of middleware and service-based integration should be coupled with a centralized data transformation engine that will further enable improved data exchange by mapping legacy application data to Core Component definitions for exchange with other systems across the enterprise. The transformation engine will also centralize and consolidate the management of data transformation logic.

To complement the data transformation engine, the internal data exchange platform will also make use of a centralized process engine to provide state management services as well as cross-lifecycle business process coordination. In addition to state management and process coordination, a centralized process engine will increase visibility and error handling for internal data exchange.

A final aspect of the recommendation is to standardize data formats based upon Core Component definitions. It is recommended that key data exchanged between internal FSA systems meet Core Component format and content standards. Initially, all file formats must be supported though the use of centralized data transformation logic; however, data should evolve to use XML and Core Component standards in the future. The Core Components will described in detail in the Core Component Dictionaries (123.1.15) document.

[2] © Copyright Gartner, Inc. Source: "Introduction to Service-Oriented Architecture", 14 April 2003.

The following diagram represents the components of the overall Internal Data Strategy recommendation for FSA:

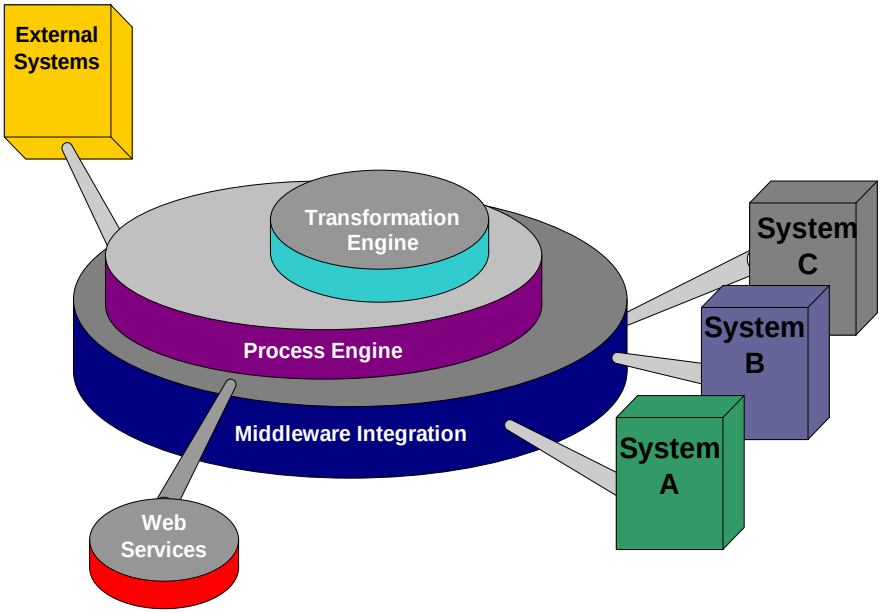

Figure 5 - Recommended Internal Data Strategy

The approach to Internal Data Strategy at FSA is phased with the intent that the overall internal data exchange architecture will evolve over time as the FSA internal data exchange environment matures. At its present state, internal data exchange at FSA takes place primarily via point-to-point batch-style data transfers between two distinct systems and uses a variety of proprietary flat file formats.

The first step in evolution of internal data exchange at FSA is to migrate to more real-time data exchange that is enabled by a hub-and-spoke architecture. The hub-and-spoke architecture will be capable of handling both batch and real-time data transfers that take place via a wide range of data formats. Centralized data transformation applied at the middleware integration layer will allow for easy exchange between multiple systems. The following figure illustrates the progression that internal data exchange will take as it moves from point-to-point integration to service-based integration over time:

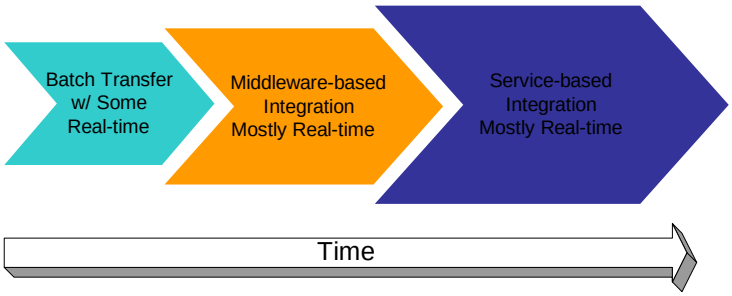

Figure 6 - Data Exchange Evolution

The following figure illustrates the evolution that data transformation will take as FSA moves from distributed transformation logic to use of Core Components across the enterprise:

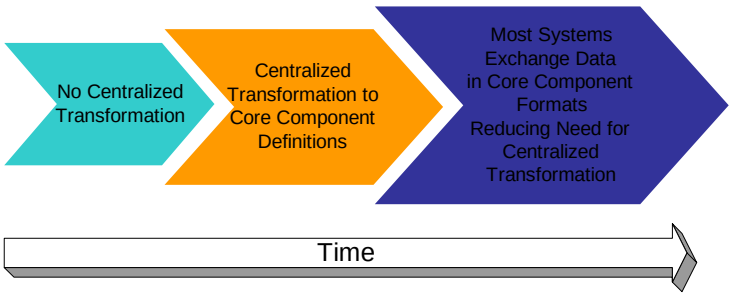

Figure 7 - Data Transformation Evolution

The next step in evolution occurs as FSA realizes increased levels of real-time data exchange and is capable of centrally managing cross-lifecycle business processes through a process engine. At this stage, centralized management of complex system interactions is realized as well as increased levels of cross-lifecycle process coordination.

The following figure illustrates the evolution of process logic application:

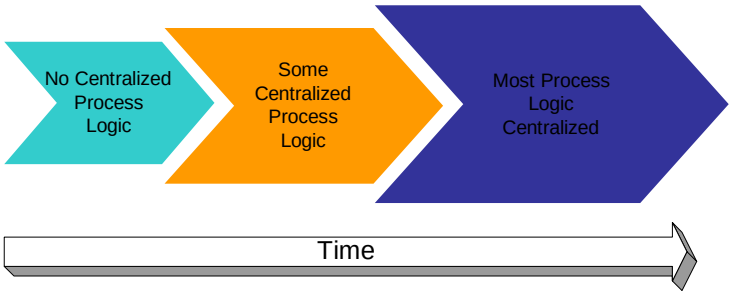

Figure 8 - Business Process Logic Evolution

As increases in both real-time exchange and centralized process management are seen, the next stage in the evolution involves migration to a more service-based architecture. At this point, like groups of services across the enterprise are grouped and centrally exposed for use by other systems within the enterprise, and the data that is used in all forms of data exchange will make use of Core Component definitions and standards.

The following figure represents the phased approach that is recommended for the application of data format standards:

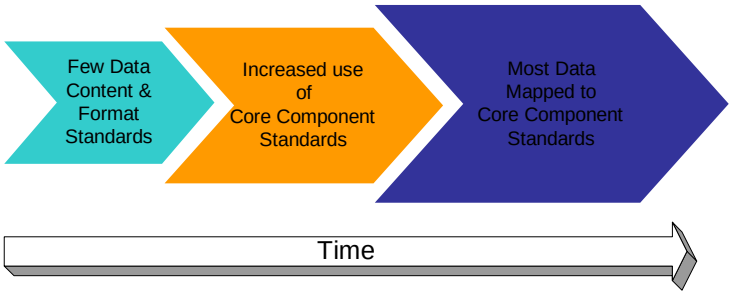

Figure 9 - Data Format Evolution

The end state in the evolution is that of complete data and service integration. At this point, access to data and services across the enterprise is relatively seamless and almost all inter-system interactions occur in real-time.

5.2 Recommendation Benefits

The foundation of the Internal Data Strategy is enablement of inter-system data exchange via a centralized hub-and spoke middleware architecture. This middleware architecture will also provide the transport framework necessary to support the evolution to service-based integration architecture in the future. The integration services are further supported by centralized process and data transformation engines, which will allow for the uniform application of inter-system process and transformation logic across all points of data exchange.

Inter-System Data Exchange Benefits

Use of a hub-and-spoke middleware platform for initial phases of internal data exchange provides the following benefits:

- Hub and spoke middleware allows for relatively easy connectivity of legacy systems that utilize batch transfer – batch transfer must be supported near term as many of FSA's internal systems currently utilize this technology.
- Connectivity for newer systems is often provided by pre-built application connectors.
- Near real-time transactions are readily supported.
- The hub and spoke middleware architecture allows for a phased implementation approach with the capability of integrating groups of like systems as it makes business sense with minimal negative impact to FSA's ongoing business.
- Hub and spoke architecture provides the foundation for enabling centralized business process logic as well as data transformation logic.

The phased migration to a service-based architecture is recommended for the following reasons:

- As more systems become enabled for real-time exchange, a natural progression to a more service-based architecture becomes possible.
- Like the earlier progression to more real-time transfers rather than batch, a phased migration approach to a service-based architecture is possible as like groups of system services can be exposed intermittently to the enterprise. A phased approach makes the most business sense for FSA while minimizing the overall impact to ongoing business.
- Utilizing the middleware transport capabilities to enable services provides guaranteed levels of message delivery that are not readily available when utilizing Internet transport protocols.
- A service-based architecture is capable of utilizing centralized data transformation and business logic facilities that exist from the earlier implementation of a hub and spoke middleware architecture.
- Over time, the service-based architecture will effectively eliminate barriers that exist between systems that currently serve different phases of the student aid lifecycle thus enabling more efficient data exchange.

Data Transformation Benefits

The centralized approach to data transformation provides the following benefits to FSA:

- Centralized transformation to Core Component definitions will further enable data exchange between multiple FSA internal systems.
- Transformation logic can be applied in a phased approach to support the overall systems integration approach.
- Data transformation logic will readily enable existing legacy applications to exchange data with other systems across the enterprise.
- System owners will be able to convert to data exchange formats that utilize Core Component formats and standards as it makes business sense.

Business Process Management Benefits

The benefits to FSA of enabling a centralized business process engine include:

- Enabling centralized management of cross-lifecycle business processes.
- The ability to route data to multiple internal and external systems based upon state.
- Enabling the ability to trace data through various lifecycle and system touch points.
- Centralizing business process coordination will allow for more efficient data exchange between internal systems across the FSA enterprise.
- Integrating external data exchange points into the overall internal data exchange business process via a centralized process engine.

Data Format Benefits

The benefits of utilizing standardized Core Component definitions for data format and content standards are:

- Ultimate adoption of Core Component and XML file formats will further enable data exchange between internal FSA systems.
- The overall integration strategy takes into account that data transformation will be required initially as many legacy systems will not immediately be able to generate files based upon Core Component or XML file formats.

5.3 Bridging of Gaps

The following table illustrates how this overall recommendation fills the gaps as derived from the overall internal data exchange business objectives:

Business Objective Rank	Gap Description	Fulfilled by Recommendation	Explanation
1	Limited business owner and user visibility exists for transactions that utilize the FSA EAI architecture.	✓	Process engine and middleware-based integration platform will allow for increased visibility and history of transactions.
	No managed workflow insight into inter-system communications.	✓	Process engine will provide for insight, management, and control of some inter-system communications/logic.
	Business logic and data transformation not centralized.	✓	Process and Transformation engines enable the centralization of inter-system logic.
2	Common identifiers for students and schools not utilized across the enterprise.		Internal Exchange may be an enabler for transport of these identifiers, but will not enforce, define or manage their use.
	Common data format does not exist.	✓	Core Components will serve as the basis for data format and content standards.
3	Legacy systems are not real-time enabled.	✓	Legacy functionality may be "wrapped" and presented as services.
	Majority of internal data exchange uses batch technologies.	✓	Outlined recommendation allows systems the ability to iteratively move towards real-time exchange when business value is achieved.
	Large, flat files do not readily enable real-time data exchange.	✓	Use of Core Components and a real-time enabled architecture allows design considerations towards real-time capabilities.

Business Objective Rank	Gap Description	Fulfilled by Recommendation	Explanation
4	Data is not available to track customer status.		Internal Data Recommendation provides a process engine that can be leveraged to ensure student information is routed appropriately from all phases of the lifecycle. The Data Storage, Access and Management recommendation will build upon this capability and outline where customer status information will reside.
	Architecture is not in place to view customer service data.	✓	Internal Data Exchange will enable accurate and visible feeds to enable customer data..
	Clearly defined cross-lifecycle business processes do not exist.	✓	Process Engine will enable the definition of business processes across multiple phases of the student aid lifecycle.
5	Duplicate data transfers occur between systems.	✓	Middleware based integration will allow for increased data sharing between multiple systems and consolidation of interfaces.
	Manual processes prevent consolidation and reuse of FSA services.	✓	Process Engine will enable the ability to automate processes across the enterprise.
	Limited reuse of services exists across the enterprise.	✓	Service-based aspects of integration platform will allow for common services to be reused by systems.

Table 22 - Recommendation Gap Bridging

5.4 Alignment with Business Objectives

Fundamentally, technology that improves processing makes sense. However, it is the realization of business value that supports change. Below is a matrix that reviews each of the previously defined FSA Internal Data Business Objectives. Assessment of the gaps should help ensure that the business objectives are reasonably met, but a specific review of the business objectives is outlined below to ensure the true business needs are enabled by the strategy recommendations. Explanations of the assessment criteria can be found in Appendix A: Business Objective Accommodation Criteria.

Rank	Business Objective	Objective Fulfilment	Explanation
1.	Centralized visibility and data flow control of the end-to-end interface		• Middleware/Service Architecture provides centralized data transport.

Rank	Business Objective	Objective Fulfilment	Explanation
	process.		• Centralized process engine enables transaction tracing. • Cross-lifecycle business process management enabled.
2.	Establish common identifiers to enable a reduction of internally exchanged and commonly referenced data.		• Common identifiers are directly addressed by the SSIM and RID efforts. • Internal Data Strategy provides the architecture for the exchange of common identifier data. • Core components simplify data transfer and allow identification of extraneous or duplicate data.
3.	Provide Right Time Processing for Internal system communications.		• Integration platform allows increased ability to provide real-time data exchange. • Enables movement away from batch transport. • Movement to Core Components and when appropriate XML allow for more concise records and facilitate real-time transactions.
4.	Ability to share customer's status throughout all phases of the lifecycle.		• Integration platform provides the architecture for transport of customer status data. • Process engine provides the ability to trace customer status across multiple phases of the lifecycle. • The Data Storage, Access and Management Strategy needs to ensure the data is available to enable this business objective – while the internal data strategy provides the ability track, transport, and act upon information.
5.	Consolidate Interfaces and provide reuse of services across the enterprise.		• Centralized integration enables the consolidation of interfaces across the enterprise (inter-system business logic, transformation, error handling etc…) • Migration over time to a service-based architecture will provide for capability re-use.

Table 23 - Recommendation Business Objective Fulfillment

5.5 Business Process Example – ISIR Production

Regardless of the excitement surrounding a new technology, a technical solution is ultimately judged on the business value it provides. To afford a better understanding of the implications related to the above recommendation, the ISIR process is reviewed for both current and a potential future state for two scenarios: 1) an error case and 2) a change to the business process. This review utilizes a "real-world" example to evaluate the Internal Data Strategy recommendation.

The following sections provide insight into how the Internal Data Strategy could enhance the ISIR production process that currently occurs at FSA. This example case validates concepts outlined in the Internal Data Strategy. A detailed evaluation of the process would need to occur before any implementation recommendations could be substantiated.

5.5.1 Current ISIR Production Process

The current ISIR production process involves both real-time and batch requests with interaction between multiple internal and external systems. As is the case with most internal data exchange that takes place at FSA, the ISIR production process utilizes point-to-point batch-driven interfaces, and the transfers are managed independently by the initiating systems.

The current ISIR production process is visually represented below:

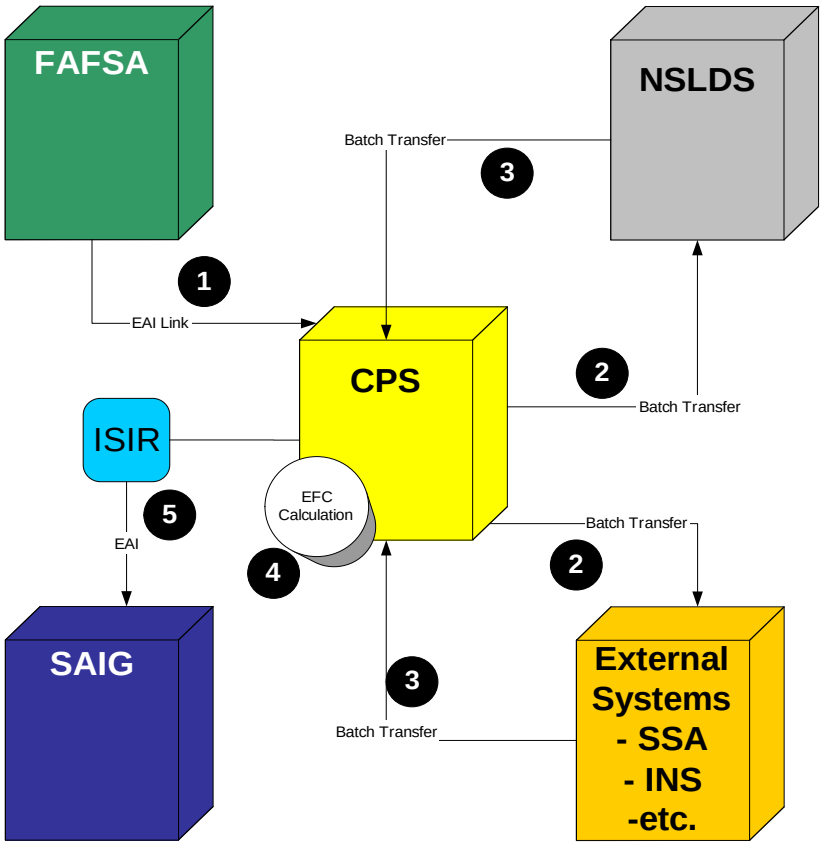

Figure 10 - Current State of ISIR Production Process

Current ISIR Production Steps:

1. FAFSA sends data to CPS via EAI.
2. CPS sends data to NSLDS via batch transfer.
2. CPS simultaneously sends requests to external systems (SSA, INS, etc).
3. Response back from external systems to CPS.
3. Response back from NSLDS to CPS.
4. CPS applies the EFC Calculation based upon data received from NSLDS.
5. ISIR is generated and sent to SAIG for distribution to schools.

Scenario 1*: An error is discovered in the ISIR file sent to schools – inaccurate data.*
In the current state, the cause of the error is not clear; therefore, data sources throughout the process must be reviewed. All data flows would need to be reviewed working in reverse from the school end to understand if a data transfer failure occurred and at what point the data integrity was compromised. At the highest level, one would need to ensure all the process steps executed successfully, and then review the specific input data at each interface in the process.

Scenario 2: *The feed from NSLDS changes and the data format is refined*
In the scenario, both CPS and NSLDS will need to change their processing. Additionally, the interface may need to be refined to accommodate the changes in format and purpose if the change is drastic. CPS and NSLDS would work together to ensure the transfer, format and logic were properly synchronized. Existing transport capabilities would be leveraged to ensure message delivery.

5.5.2 Proposed ISIR Production Process

In the proposed process for ISIR production that is based upon the Internal Data Strategy recommendation, all of the point-to-point batch-style interfaces are replaced with connections to the middleware-based integration platform. Each piece of data will be sent through the middleware-based integration platform and then routed to the appropriate destinations at the correct time by the centralized process engine. Centralized data transformation services are also provided to enable mapping from system specific formats to a common format when needed.

The new process for ISIR production is visually detailed below:

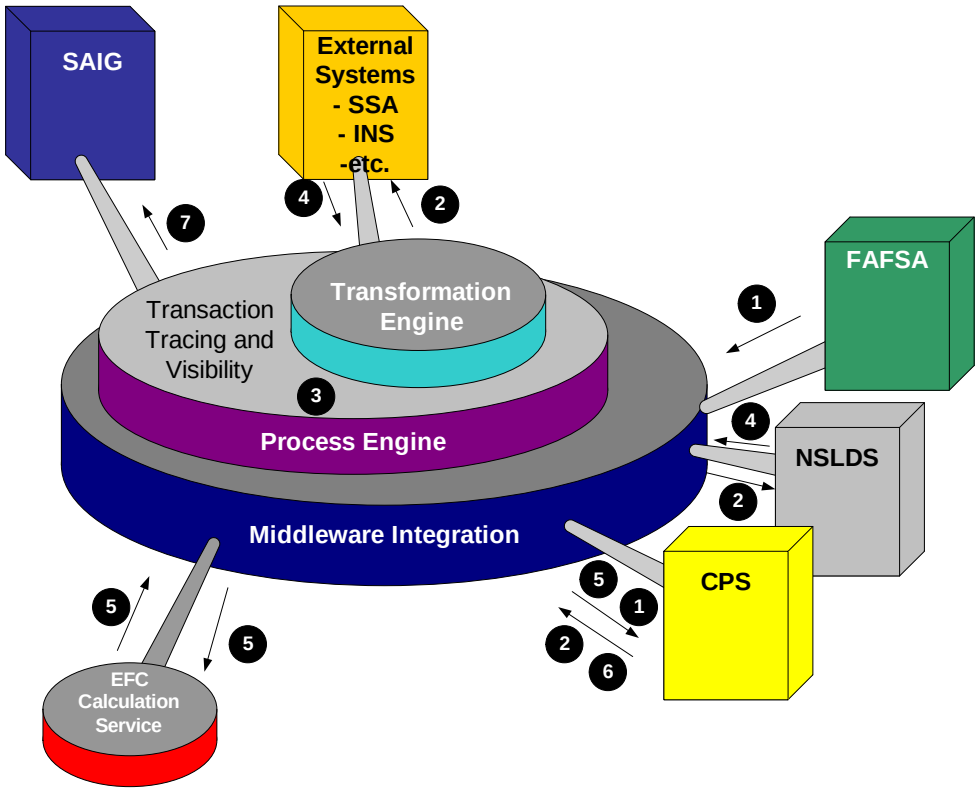

Figure 11 - Proposed ISIR Production Process

Proposed ISIR Production Steps:

1. FAFSA sends information routed to CPS via integration platform.
2. CPS sends requests to external partners and NSLDS via integration platform.
3. State of requests managed by centralized process engine.
4. Data returned to integration platform by external partners and NSLDS.
5. Process engine routes back to CPS and initiates web service call to EFC Calculator.
6. CPS distributes ISIRS to integration layer.
7. Integration layer routes and manages distribution to schools (via SAIG).

Scenario 1: *An error is discovered in the ISIR file sent to schools – inaccurate data.*
A complete history of the data transfers between both internal and external data transfers are held in the process engine. This history would be reviewed to understand if there were any points of failure that may have affected data accuracy. Specific transfers or systems feeds would be evaluated for data inaccuracies based upon the review of transfer history.

Scenario 2: *The feed from NSLDS changes and the data format is refined*
In the scenario, the data format and feed changes are addressed separately. Since transformation is centralized, modifications to the feed can be considered for processing. If this transformation already exists across the enterprise, logic can be reused and simply applied to the transfer enabling a quick implementation. If the feed changes drastically, the process engine

could apply criteria or validation where necessary. The overall ISIR process model would be updated to handle the changes in data feeds and process logic.

As seen in the scenarios above, the recommended solution allows for a more flexible architecture and provides the following key benefits:

- Error handling is done at a business/process level initially as opposed to reviewing detailed transfer points.
- Changes in format and process are centrally applied, which minimizes the amount of collaboration required between parties and reduces number of system interface changes required.
- Level of effort to accommodate new/tombstone systems is reduced due to the loosely coupled integration of interfaces.
- Multiple point-to-point interfaces are replaced by single connections to the integration platform enabling data exchange between any number of systems and reducing management complexity.
- Process engine improves the internal user experience through the capabilities for state management as well as centralized error handling and reporting.
- Process engine allows business owners to define, create, and modify business process steps across the enterprise.
- Data exchange and transformation to common formats is enabled to help to improve data quality and simplify data exchange between systems.
- The EFC Calculator is exposed as a service across the enterprise resulting in standardization and potential for reuse.

Appendix A: Business Objective Accommodation Criteria

The table below provides a rating scale that explains the measures and criteria used to evaluate how well a particular technical solution satisfies FSA's business objectives. These indicators appear in the Executive Summary and Recommendation sections with respect to both the current and the recommended states for the Internal Data Strategy.

Rating Indicator	Synopsis	Criteria
	Fully Accommodated Objective	The business objective is being completely satisfied.
	Well Accommodated Objective	The business objective is mostly satisfied.
	Partially Accommodated Objective	Some business objective criteria in place, but the business objective is only partially fulfilled.
	Minimally Accommodated Objective	Few parts of the business objective are satisfied.
	Capability Not Accommodated	The solution does not have this element in place and the business objective is not met.

Table 24 - Business Objective Accommodation Criteria

Appendix B: Meeting Minutes – Working Session #1

Date: Monday, 07/14/2003

Time: 1:00 pm until 3:00 pm

Location: 221A

Objectives: The objectives of the first working session were to review business objectives and gaps, understand key decision points, establish assessment criteria, and refine options to satisfy business objectives for Internal Data Exchange.

Attendees:

Name	Business/System Area	E-Mail	Phone (Work)	Attendance
Mike Giordano	CSC	Mgiorda4@csc.com		Phone
Terry Hardgrave	Pearson	Terry.hardgrave@ed.gov	202.377.3238	X
Terry Helwig	Integration Partner	Terry.helwig@accenture.com	202.962.0792	X
Denise Hill	FSA/CIO	Denise.hill@ed.gov	202.377.3030	X
Corwin Jennings	Ombudsman	Corwin.jennings@ed.gov	202.377.3291	X
Bruce Kingsley	Integration Partner	Bruce.kingsley@accenture.com	202.962.0793	X
Robert Laurence	DTI/Students	Robert.laurence@ed.gov	202.377.3389	X
Davis Peden	Integration Partner	Davis.w.peden@accenture.com	202.962.0651	X
Kathryn Pirnia	FSA EITM	Kathryn.pirnia@ed.gov	202.377.3558	X
Harris Sibunruang	Integration Partner	Harris.s.sibunruang	202.962.0789	X
Dwight Vigna		Dwight.vigna@ed.gov		Phone
Christine Williams	FSA/EITM	Christine.williams@ed.gov	202.377.3571	X
Marty Winslow	EDS	Marty.winslow@ed.gov		Phone
Terry Woods	FSA/CIO	Terry.woods@ed.gov	202-377-3023	X

Action Items from Previous Meeting Minutes
N/A

Discussion Points
The meeting opened with a review of the status with respect to the Technical Strategies Roadmap. The meeting agenda and working session goals were also covered.

A question was raised as to what owner and user visibility means.

A point was raised that a "Central Data Router" which provides data visibility and touchpoint data currently exists. Any new capability should enable the services that it currently provides and then build upon it.

DECISION POINTS DISCUSSION

A question was raised as to how many data stores are currently hit in order to fulfill the average request. The Data Framework team will be able to answer this. It was mentioned that the possibility of using 'virtual databases' should be considered.

Concern was raised over the prioritization of business objectives. The group recognized that if the objectives are no considered holistically early on, the final recommendations for each technical area could be skewed. It was clarified that the purpose of the final deliverable (123.1.12 - Technology Vision and Strategic Plan) is to address all technical areas.

ASSESSMENT CRITERIA DISCUSSION

A point was raised that there might be a need for assessment criteria that addresses 'value'. Resolution – a list of benefits associated with each option will be included.

A question was asked as to how 'level of effort' would be gauged. It was decided that 'level of effort' would be ranked on a "low, medium, high" scale.

A point was raised that the maturity level of each piece of technology should be considered.

Are there examples of every technology that is being used in mission critical situations? Risks based on technology maturity and usage must be considered.

It was emphasized that 'bleeding edge' technology often carries a negative connotation for business owners. Business owners are only concerned with technology that will help them meet their needs.

MESSAGE ORIENTED ARCHITECTURE DISCUSSION

It was clarified that Message Oriented Middleware platforms are capable of providing process logic, transformations logic, etc. in addition to basic message transport functions.

It was suggested that a current underlying problem is that business logic is spread across multiple applications with different applications often performing similar functions in different ways. It was suggested that some business logic in addition to inter-system business logic should be centralized as well.

It was clarified that the EAI architecture in place today performs tightly coupled point-to-point transfers with the capability to exchange data with multiple interfaces. The PEPS data file is currently being shared with COD and EZAudit.

SERVICE ORIENTED ARCHITECTURE

Inquiry was made as to what the official definition of a Web Service is. The group felt that care should be taken not to tie Service Oriented Architecture to a particular transport type (e.g., Web services).

Concern was raised over many FSA internal data transfers taking place over WAN technology. Performance needs to be taken into consideration before technology is determined. Any performance standards that must be met today, but also apply to new architectures.
It was agreed that Web services would not be considered the only enabler of a service-oriented architecture.

Concern was raised over the maturity of Web services and its ability to perform functions surrounding security, batch transport, etc.

ADDITIONAL CONSIDERATIONS
Based upon discussions, it was decided that the categorization of options did not fully meet FSA's needs. The following matrix was created to outline the various technology options available and provide a logical means for grouping:

Option 1: Point-to-Point, MOM, centralized transform, centralized business process

Option 2: SOA, Leverages Web services, Decentralized Transformation, Decentralized Logic

Option 3: MOM and WS, Mix of Transformation, Mix of Logic

	Transport Aware	Transport Unaware
Architecture Type	Point-to-point[1] Traditional EAI	Service Oriented Architecture[2]
Transport Layer	Message Oriented Middleware[1] FTP	Message Oriented Middleware Web services[2] Both[3]
Transformation	Centralized[1] Decentralized	Centralized Decentralized[2] Mix[3]
Business Process Logic	Centralized[1] Decentralized	Centralized Decentralized[2] Mix[3]

New Action Items

Action Item	Owner	Date Due
Options for each category should be sent to Kyle Michl and Denise Hill prior to the meeting on Wednesday.	Everyone	7/16/2003

Next Meeting Time
Wednesday, July 16 – 10am.

Suggested Agenda Items
None at this time.

Appendix C: Meeting Minutes – Working Session #2

Date: Wednesday, 07/16/2003

Time: 10:00 am until noon

Location: 221A

Objectives: The objectives of the second working session were to review options for different components that make up the Internal Data Strategy and to decide upon a combination that will best meet FSA's future needs.

Attendees:

Name	Business/System Area	E-Mail	Phone (Work)	Attendance
Mike Giordano	CSC	Mgiorda4@csc.com		Phone
Roger Hartmuller	Integration Partner	Roger.l.hartmuller@accenture.com	202-962-4160	X
Terry Helwig	Integration Partner	Terry.helwig@accenture.com	202.962.0792	X
Denise Hill	FSA/CIO	Denise.hill@ed.gov	202.377.3030	X
Paul Hill, Jr.	Title IV	Paul.hill.jr@ed..gov	202.377.4323	X
Holly Hyland	Title IV	Holly.Hyland@ed.gov	202.377.3710	X
Corwin Jennings	Ombudsman	Corwin.jennings@ed.gov	202.377.3291	X
Bruce Kingsley	Integration Partner	Bruce.kingsley@accenture.com	202.962.0793	X
Robert Laurence	DTI/Students	Robert.laurence@ed.gov	202.377.3389	X
Davis Peden	Integration Partner	Davis.w.peden@accenture.com	202.962.0651	X
Kathryn Pirnia	FSA EITM	Kathryn.pirnia@ed.gov	202.377.3558	X
Harris Sibunruang	Integration Partner	Harris.s.sibunruang	202.962.0789	X
Dwight Vigna		Dwight.vigna@ed.gov		Phone
Marty Winslow	EDS	Marty.winslow@ed.gov		Phone
Terry Woods	FSA/CIO	Terry.woods@ed.gov	202-377-3023	Phone

Action Items from Previous Meeting Minutes
Options for enabling internal data exchange were sent to Kyle and Denise. These options were then consolidated and pieces were used to facilitate this meeting.

Discussion Points
The meeting opened with a re-cap of Monday's first working session.

The approach to this working session is to break-up the overall options that were presented in the first working session into a series of manageable questions that address distinct technological pieces.

It was suggested that we attempt to leverage solutions that are already in place for inter-system connectivity rather than build "new" solutions.

BATCH TRANSFER DISCUSSION

A question was raised as to whether or not customized forms of batch transfer currently take place. It was clarified that most of the batch transfer that takes place occurs via proprietary interface.

It was pointed out that a custom package for large batch transfer currently exists – Data Preparation Architecture software.

It was agreed upon that point-to-point forms of middleware transport should be used to connect two systems that need to communicate with each other with enhanced levels of delivery.

It was emphasized that hub-and-spoke middleware architecture will transport information to a central location before it is shared with multiple systems. Middleware architecture will allow for data reuses and increased visibility among systems.

It was clarified that services over middleware will enable business functions that exist within specific systems across the enterprise. These services will typically involve request/reply transactions.

A question was raised as to whether or not large amounts of data would be moved over middleware architecture. It was clarified that middleware would be the desired method of transport due to guaranteed levels of delivery, etc.

A concern was raised over the hub-and-spoke architecture possibly being a single point-of-failure for data exchange. It was emphasized that architectural decisions will need to be made regarding failover, backup, etc.

Concern was raised over the impact of sending large volumes of batch data across the network– specifically when data is being shipped across different geographies.

The group agreed that middleware hub-and-spoke is the best approach for point-to-point system connectivity.

BATCH TRANSFER DATA TRANSFORMATION

A question was raised over how much the amount of data transformation logic would increase over time. Initially, data transformation logic will be required to integrate legacy systems. As systems begin to exchange data in a "common language", less data transformation logic will be required.

The group agreed that centralized transformation logic would be used if data is being transferred to multiple systems.

A question was raised as to whether or not a "data store" would be associated with the centralized transformation logic. It was clarified that the transformation engine would only contain rules and transformation logic – no business data would be stored here.

Concern was raised over system-level expertise and business logic being pushed into the middleware layer. It was clarified that inter-system business logic would be the only logic that is located at the middleware layer – application-specific logic would remain within each application.

Data transformation capabilities will be used to make common data definitions across the enterprise more of a reality.

A question was raised as to what format systems should use when placing data into the middleware. It was clarified that ideally, the format should be that of Core Components, but transformation logic would be required to enable legacy exchange.

CENTRALIZED BATCH MANAGEMENT

A question was raised as to why a process engine was needed to enable end-to-end visibility. It was clarified that end-to-end visibility was a capability that FSA enjoyed when the entire student aid process was internal. It was also added that cross-lifecycle business process management was a desire of business owners as well.

It was clarified that a process engine would be centrally managed, but not every piece of data would have to go through the engine. Centralization would only occur where needed.

REAL-TIME TRANSFER DISCUSSION

A question was raised as to what the exact definition of real-time data exchange is. It was clarified that real-time data exchange incorporates both real-time and near real-time exchange. There may be some transactions where some amount of latency would be acceptable.

Another option to consider is direct database-to-database connectivity. Here PL/SQL scripts or custom applications would handle the data exchange.

A question was raised over what kind of delay could be expected for service-based calls over a middleware platform. It was clarified that this was a detailed design consideration.

The group decided that the best approach to real-time exchange was to initially leverage a hub-and-spoke middleware platform, but to ultimately move to service-based where possible.

REAL-TIME TRANSFER DATA TRANSFORMATION

The group felt that the same approach as used for Batch transfer transformation could be used for real-time; therefore no additional discussion was required.

CENTRALIZED REAL-TIME MANAGEMENT

The group noted that criteria must be established for real-time process management. As more transfers occur via real-time means, there are potentially more integration points to manage.

A good example of a business process that would benefit from centralized process management is the updating of student information across all systems. If an update were to happen on one system, centralized process management could dictate that the same change occur on all systems.

For service-based exchange, the group decided that centralized means of monitoring service state is necessary.

KEY DECISION SUMMARY

No.	Key Decision(s)	Options	Potential Solution
1.	a. How will batch transfers occur?	• Middleware HUB and Spoke • Middleware Point to Point • Service Based request over middleware • Custom	Mix of hub and spoke middleware and service-based exchange.
	b. For Large Legacy to Legacy Transfers?	• Middleware HUB and Spoke • Middleware Point to Point • Service Based request over middleware • Custom	Hub and spoke middleware.
	c. For future batch based systems?	• Middleware HUB and Spoke • Middleware Point to Point • Service Based request over middleware • Custom	Hub and spoke middleware trending towards service-based integration.
2.	a. Will transformation need to occur on batch transfers?	• Yes – • No • Sometimes	Sometimes, only when data is exchanged with multiple systems.
	b. How will batch transfers that require transformation be handled?	• Centralized - capability executed and managed from common location • Decentralized – individual systems own and manage independently • Mix (apply criteria)	Data transformation logic will be centrally applied as needed.
3.	Do we need to centrally manage/provide visibility to the batch transfers between systems? How?	• Process engine that controls/manages system to system transfer • Middleware Pub/Sub • Custom	Centrally apply process logic when it makes sense. First get data touch points and then enable visibility.

No.	Key Decision(s)	Options	Potential Solution
4.	a. How will real-time transfers be managed between systems? (small data)	• Middleware Hub and Spoke • Middleware Point to Point • Direct Database • Services leveraging middleware • Services leveraging inter/intranet • Mix (apply criteria)	Mix of point to point and hub and spoke middleware.
	b. Real-time to Legacy?	• Middleware Hub and Spoke • Middleware Point to Point • Direct Database • Services leveraging middleware • Services leveraging inter/intranet	Mix of hub and spoke middleware migrating to service-based integration.
	c. Real-time to future systems?	• Middleware Hub and Spoke • Middleware Point to Point • Direct Database • Services leveraging middleware • Services leveraging inter/intranet	Mix of hub and spoke middleware migrating to service-based integration.
5.	a. Will transformation need to occur on real-time transfers?	• Yes • No • Sometimes	Sometimes, only when data is exchanged with multiple systems.
	b. How will transfers that require transformation be handled?	• Centralized - capability executed and managed from common location • Decentralized – individual systems own and manage independently • Mix (apply criteria)	Data transformation logic will be centrally applied as needed.
6.	Do we need to manage/provide visibility to the real-time exchange of data between systems? How?	• Process engine that controls/manages system to system transfer • Middleware Pub/Sub • Custom	Centrally apply process logic when it makes sense. First get data touch points and then enable visibility. Use with service-based exchange to monitor services.

New Action Items

Action Item	Owner	Date Due
None identified.		

Next Meeting Time

To be announced.

Suggested Agenda Items
None at this time.