# Delivery Vehicle Definition

This deliverable describes the end-to-end technology components required to deliver the service characteristics for a user service class. This includes hardware and system software as well as the execution, operations, and development environments. This definition is used extensively throughout technology infrastructure planning. The deliverable is the starting point for the Delivery Vehicle / Service Matrix that defines the delivery vehicle in more detail.

| | | |
|---|---|---|
| **I.** | **IPT Name:** | |
| **II.** | **Deliverable Name:** Delivery Vehicle Definition | **Date Completed:** |
| **III.** | **Contact Information** | |
| | Name | Channel Unit |
| IPT Sponsor | | |
| Channel Task Manager | | |
| CIO Task Manager | | |
| Contractor Task Manager | | |
| **IV.** | **Task Order Number:** | |

# Description

This document details one particular example of a delivery vehicle and its key attributes in each of the following characteristics:

- Description
- Use Case
- Functional Attributes
- Performance Attributes
- Interface Attributes
- Logical Specifications

**This is the Delivery Vehicle sample example.**

## 4.4.3 Web Services

### 4.4.3.1 Description

The Web Service provides the required infrastructure to support applications which impose minimal demands on intelligent workstations through the use of common sponsoring organization technology, and a standard (published) application level protocol which mediates the level of process / data distribution. Examples of such technologies today include X-Windows and WWW technology, with the latter providing the opportunity for dynamic process distribution through runtime environments such as Java. In the future, the same concepts embodied by these examples should be preserved and enhanced with greater functionality, while preserving the essential quality of low demands on the client workstation. The defining element of this Delivery Vehicle is that it has "near ubiquitous" access potential.

The Web Service Delivery Vehicle will enable the deployment of applications and information to a much broader reach of end users by allowing solution providers to use a "lowest common denominator" access approach. Implications are that end users which might not normally be provisioned with an application / information support tool might now be reachable, leveraging cost / benefit assessments by lowering workstation infrastructure costs.

- Workstation is provisioned with "least common denominator" software (Ultra Thin Client Program) which allows access to network applications and data. This would include network access software, application/remote data access software, and optionally, local data storage capability. Ideally, commonly available network access should be assumed (i.e. PRN, WWW, public network dial-up access would be examples today, with increasingly higher opportunity for solution deployment).
- A Device Intelligence Layer allows non-intelligent devices to be integrated to the solution. Computer Telephony Integration and bi-directional facsimile handling are two examples which could be deployed using such a layer.
- The Ultra Thin Client Program needs to be able to accept and run "Imported" programs, obtained dynamically from external sources. Today, JAVA applications exemplify this - tomorrow, brokered environments could dynamically "decide" on the distribution strategy for a particular user/process. For example, if User A uses a mail program every day, the brokered environment could "decide" that the best location for the mail program is the user's intelligent workstation. If, after a while, User A stops using the mail program (perhaps because a better one has come along), the brokered environment can recognize this and clear (delete) the program from User A's station.
- The interaction is graphical based. Utilized when the emphasis is on random events at the expense of response times.
- Processing is distributed between systems, and can change dynamically over time, as required by the technology solution.
- Enables user driven business processing at locations which may not be cost-effective to either conduct application/data distribution or to provide enhanced network access.
- Provides the opportunity to enhance non-intelligent access devices by "adding intelligence" to them. For example, a telephone could use an Web Service solution if the solution is front-ended with a Computer Telephony Integration component.

- Can access and present multiple data storage types (RDBMS, image, sound, etc...).

**4.4.3.2 Use Case**

As an example of Web Service usage, a potential future offering will be enabled: suppose that the XYZ were to offer an e-mail post office box to every citizen and business for a fee. The fee would have to be small enough to compete with other similar offerings in the marketplace, and perhaps even be offered "free of charge", with revenue coming from other aspects of the service. A cost / benefit analysis of provisioning alternate solutions which would require users to obtain XYZ-specific software program, and for the XYZ to offer a wide breadth of variants of the program (NT, Win 95, Mac, etc...) would not necessarily support a business case for the new service offering. On the other hand, the use of a "ubiquitous" platform (the exact composition of which will change over time) as the deployment infrastructure for the offering would greatly improve the economics - the cost of deploying the solution would certainly drop, as it would eliminate XYZ responsibility for deployment and management of a large portion of the solution infrastructure. The e-mail post office offering could be provided via the "ubiquitous" platform of the day (currently the WWW and telephone networks), and could take advantage of emerging technology components such as 2-way facsimile to accept and send e-mail messages. End users would not need to be concerned about XYZ-specific access technology to use the new service.

#### 4.4.3.3 Functional Attributes

- Utilizes “least common denominator” infrastructure available to a large number of potential users.
- Provides enterprise and (if desired) public-wide access to information.
- Dynamic allocation of application/data in accordance with solution requirements.
- Supports mobile access.
- Supports remote access.
- Supports full-text search engine.
- Provides graphic support.
- Works within the constraints of the XYZ security environment.

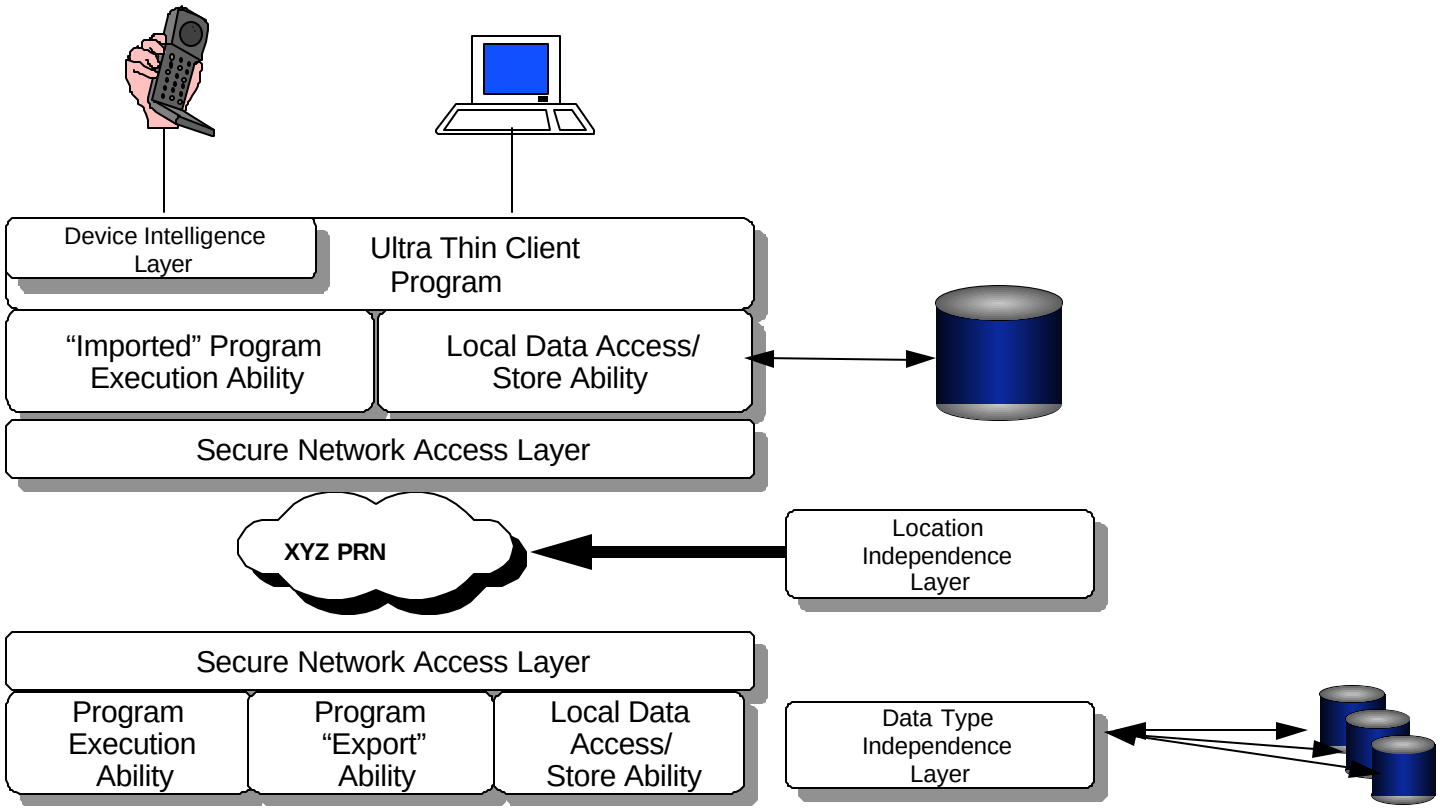
#### 4.4.3.4 Performance Attributes

- Supports 7x24, as access characteristics are not predictable/controllable. Scheduled downtime of service components should not be visible to the end user, as this defeats the “ubiquitous” nature of the offering.
- Reliability.
- Interruptions greater than one hour are assumed to critically impact business.
- Performance is not critical to business. This is dependent on the infrastructure access of the end user. The “least common denominator” access type provides industry-accepted standard performance. For example, if in the year 2000, 56Kb access per user is generally accepted for the target end user audience for the solution which uses Delivery Vehicle, this is the “least common denominator” which the service needs to meet.
- Benchmark for “ubiquitous-ness” is the U.S. marketplace. Over time, this could evolve to include other jurisdictions, as dictated by the business needs of the XYZ.

#### 4.4.3.5 Interface Attributes

- Meets XYZ data security policy standards.
- Graphical User Interface.
- Supports XYZ database standards (e.g., Oracle and SQL).
- Meets XYZ security environment standards.

#### 4.4.3.6 Logical Specifications


Device Intelligence Layer
Ultra Thin Client Program
"Imported" Program Execution Ability
Local Data Access/ Store Ability
Secure Network Access Layer
XYZ PRN
Location Independence Layer
Secure Network Access Layer
Program Execution Ability
Program "Export" Ability
Local Data Access/ Store Ability
Data Type Independence Layer