

Systems Development Interface Control

An interface, from the perspective of system development, can be identified as any point where a system and something, or someone, meet. This interface point may include other systems, internal hardware, circuitry, external peripherals, networks, system users, etc. For example, common interfaces for computer peripherals may include USB, serial, parallel ports, etc; common interfaces for system users may include monitors, keyboards, mice, etc.

Interfaces are documented using interface control documents (ICD) that describe the system's interfaces as well as any rules for communicating with them. ICDs help ensure compatibility between system segments and components and are key elements of effective system design. The purpose of the ICD is to clearly communicate all possible inputs and outputs from a system for all potential actions whether they are internal to the system or transparent to system users. The level of detail included in any ICD is dependent upon the requirements of stakeholders.

ICDs define for the development team system inputs, outputs, internal interfaces, and interfaces between other systems, subsystems and/or users. An ICD should only describe the interface itself and not any characteristics of the systems using it. An ICD should also not include anything about the meaning or intended use of the data. Any features, functions, or logic supporting the system interface should be outlined within separate design and/or specification documents. Defining an ICD in this way allows other teams to develop connecting systems without concern of how the data is treated. This allows for modularity that leads to easy maintenance and extensibility of systems.

Depending on the specific nature of the system, it is possible that a number of different ICDs may be required, one for each different interface type. For example, network interface control document, graphical user interface control document,

application program interface document, etc. Regardless of the interface type, format, or name, an effective ICD would be used in conjunction with relevant system design and specifications documents, models, and drawings, to communicate how the overall system functions and interacts. An effective ICD may include:

- Description of interface and interface types. For system development, this may include information such as data type, size, format, measures, etc
- Description of commands and parameters that pass between systems
- Description of how systems communicate with subsystems
- Description of how systems communicate with external systems
- Description of how systems communicate with its users
- Description of how systems interprets external commands and what responses/behaviors are expected

The Federal Enterprise Architecture (FEA) Reference Model contains components that reference many of the items that should be considered when defining ICDs. Some of these items include:

- Interfaces between systems and their users
- Levels of access required by the application and other systems
- End-to-end management of communication sessions between systems and system components
- Hardware, operating system, other software, network protocols, etc that provide or request information
- Data storage, modification, and extraction of information from databases and any required interfaces
- Physical devices that provide the computing and networking within and between systems
- Security methods for protecting information and systems and how such policies/mandates may affect system interfaces
- Connections between the system, users, other systems, and peripherals
- Method for enforcing business rules



- Structure, elements, objects, and properties of data to be exchanged
- Methods in which data is transferred and represented in and between systems and how data is discovered and shared
- Communicating, transporting, and exchanging information through common dialogs or methods

An ICD also allows project teams to leverage other systems by providing developers with the information necessary to build add-on or complementary applications or to simply interface with existing systems. Some examples of interfaces where an ICD may be required include:

- *Application Layer* interfaces directly to and performs common application services for the application processes; it also issues requests to the presentation layer.
- The *Presentation Layer* establishes a context between application layer entities, in which the higher-layer entities can use different syntax and semantics, as long as the Presentation Service understands both and the mapping between them.
- The *Session Layer* controls the sessions between computers. It establishes, manages and terminates the connections between the local and remote application.
- The *Transport Layer* provides transparent transfer of data between end users, providing reliable data transfer services to the upper layers. The transport layer controls the reliability of a given link through flow control, segmentation, and error control.
- The *Network Layer* provides the functional and procedural means of transferring variable length data sequences from a source to a destination via one or more networks while maintaining the quality of service requested by the Transport layer. The Network layer performs network routing functions, and might also perform fragmentation and reassembly, and report delivery errors.
- The *Data Link Layer* provides the functional and procedural means to transfer data between network entities and to detect and possibly correct errors that may occur in the Physical layer.
- The *Physical Layer* defines all the electrical and physical specifications for the device. In particular, it defines the relationship between the device and a physical medium. This includes the layout of pins, voltages, cable specifications, Hubs, repeaters, network adapters, Host Bus Adapters (HBAs used in Storage Area Networks) and more.

For more information on project management, removing barriers to project team development, the Project Management Community of Practice or the CDC UP please visit the CDC Unified Process website at <http://www.cdc.gov/cdcup/>. ■

Project Management Community of Practice

- **July 27, 2012**
PM Best Practices – A Panel Discussion
- **August 24, 2012**
Enterprise Performance Life Cycle (EPLC)
- **September 28, 2012**
A Conversation with CDC Policy Leadership
- **October 26, 2012**
The Value of Alternative Analysis
- **December 07, 2012**
Managing Risk

For more information on the Project Management Community of Practice visit the PMCoP website at <http://www2.cdc.gov/cdcup/library/pmcp/> ■

CDC Unified Process Presentations

The CDC UP offers a short overview presentation to any CDC employee and/or contractor group. Presentations are often performed at your facility, on a day of the week convenient for your group, and typically take place over lunch structured as one hour lunch-and-learn style meeting.

Contact the CDC Unified Process at cdcup@cdc.gov or visit <http://www.cdc.gov/cdcup> to arrange a short overview presentation for your group. ■

Contact the CDC Unified Process

The CDC Unified Process Project Management Newsletter is authored by Daniel Vitek, MBA, PMP and published by the Office of Surveillance, Epidemiology, and Laboratory Services.

For questions about the CDC UP, comments regarding this newsletter, suggestions for future newsletter topics, or to subscribe to the CDC UP Project Management Newsletter please contact the CDC UP at cdcup@cdc.gov

<http://www.cdc.gov/cdcup/>

