Interface Template

	I/F Element	Description
Interface signature	Interface name	A unique identifier for the interface
	Exceptions	The name and data content for each operation’s exceptions
	Properties	The name and type of each property
	Operations	The name of each operation, together with the input and output parameters and exceptions
	Operation descriptions	Description of each operation using • informal description or • pre/post condition template • example showing typical calling usage *(optional)*
	Protocol *(optional)*	Constraints on the order in which operations may be called (Statechart)
	Service Level *(optional)*	Non-functional requirements to be met by the services provided by the interface (operations)
	Notes and Issues	List of components using I/F List of issues to be resolved

Interface Template

Interfaces

- Interfaces are the means by which components interact An interface is a list of operations providing a coherent service.

Guidelines for creating Interface Specifications

The interface signature (exceptions, properties, and operation signatures) provides the syntax of the interface. To add semantics, the following fields need to be completed:

Operation semantics: Description of each operation using

- informal text, or
- pre/postcondition template
- example showing typical calling usage *(optional)*

Interface Protocol (*Optional*): Constraints on the order in which operations may be called (Statechart). If there are no constraints on operation sequence, don't include this section.

Service Level: The service level covers guarantees regarding the qualities or non-functional requirements (such as timing constraints, CPU budget restrictions, memory restrictions, availability, mean time between failures, mean time to repair, throughput, latency, data safety for persistent state, capacity, and so on) to be met by the interface and its constituent operations.

Notes: This section collects together notes about the interface that you don't want to lose track of, such as:

- what components use this interface (*this may be important enough to make into a section in its own right*)
- ideas for the component design and/or implementation

Issues: List issue and issue owner/due date. (Project management guideline: Don't get bogged down in issues. Record issue, assign an owner and issue resolution or review due date, and move on.)

Uses of Interface Specifications

To the architecture team:

- makes the architecture precise and actionable

To component developers:

- provides the contract that states what the provider has to implement to meet the services promised by the interface.

To component users:

- provides the contract that states what the client needs to do to use the interface.