Model-Based Development of Spacecraft Onboard Functions

Takahiro Yamada and Keiich Matsuzaki

JAXA/ISAS

Agenda

1. Why do we need a model?
2. Functional Model of Spacecraft (FMS)
3. Spacecraft Monitor and Control Protocol (SMCP)
4. Spacecraft Information Base (SIB)
5. Development of Onboard Software
6. Conclusion

Why Do We Need a Model?

Present Design of Spacecraft

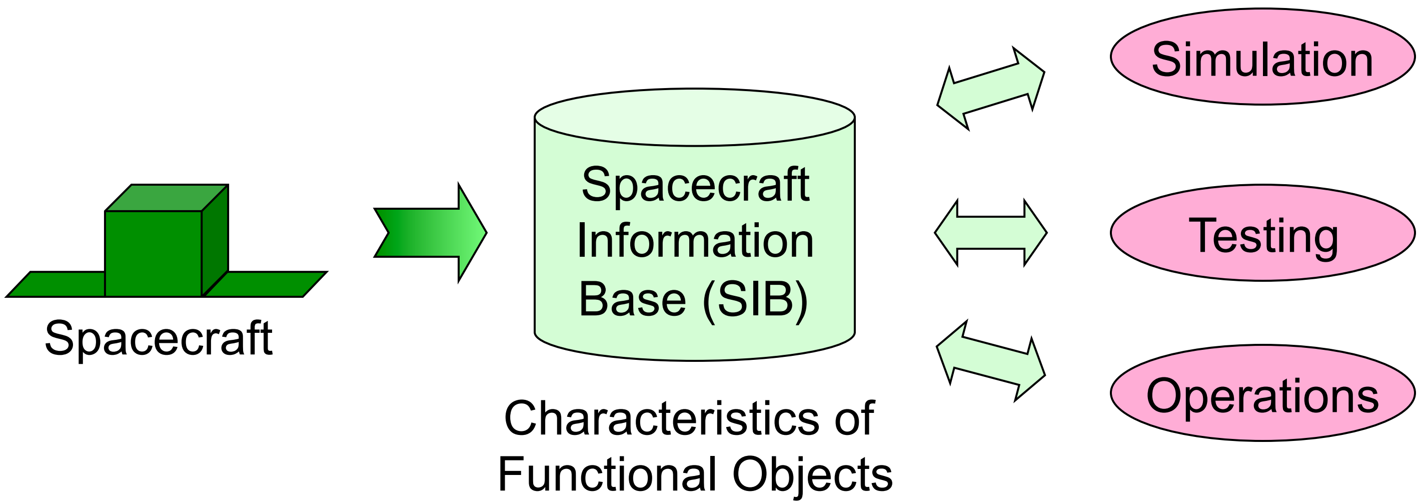
- Presently, the functional design of spacecraft onboard components is performed almost independently for each component without using any standards.
- This prevents reuse of the design of onboard components and reuse of the systems that support their development (e.g., design support systems, testing systems, operations systems, etc.) from component to component or from spacecraft to spacecraft.
- Furthermore, there are no standards on how to describe the functional design and operational rules for onboard components. Therefore,
  - Each designer has to devise a way of describing the functional design and operational rules for his or her components, and
  - People who use this information (e.g., spacecraft testers, spacecraft operators, etc.) have to learn how to read the description written by each designer.

Functional Model of Spacecraft (FMS)

- In order to solve these problems, this paper proposes a method of developping spacecraft onboard functions by using a Functional Model of Spacecraft (FMS).
- FMS provides a standard way of designing the functions of onboard components.
  - By using this model, the functional design of onboard components can be specified in a formal and unambiguous way.
  - Furthermore, reuse of functional design of onboard components will be easier because the information on components can be shared more easily among designers of onboard components.
- FMS is based on the object-oriented software technology and uses the concept of Functional Object (FO). FO is a unit for the functions of onboard components and captures information necessary for operating onboard components.

Spacecraft Information Base (SIB)

- By using a formal method for describing FOs, a standard way of storing and managing the characteristics of FOs can also be obtained.
- FMS enables development of a standard database that stores the functional information of any component of any spacecraft as the characteristics of FOs. This standard database is called the Spacecraft Information Base (SIB).



Spacecraft Monitor and Control Protocol (SMCP)

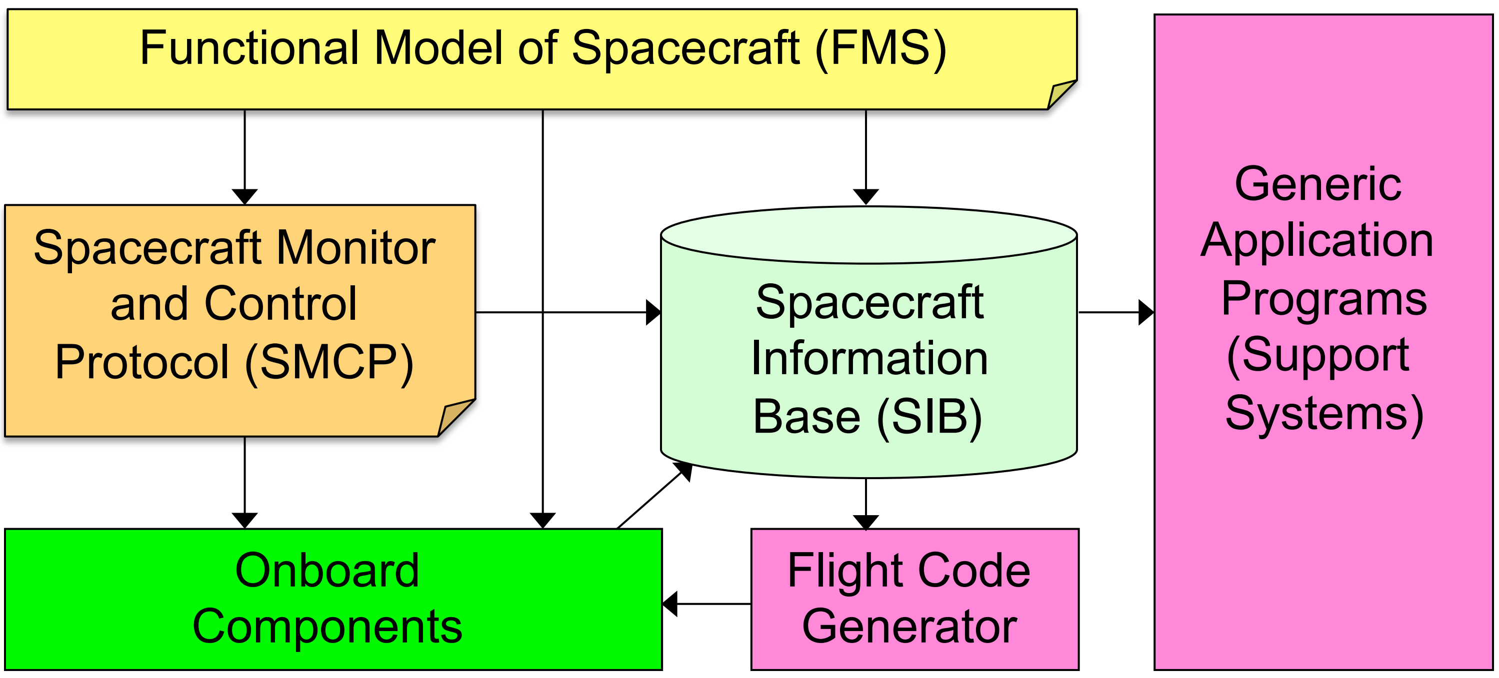
- To monitor and control onboard components from the spacecraft

  same spacecraft, command messages are sent to the components and telemetry messages are generated by the components.
- The Spacecraft Monitor and Control Protocol (SMCP) specifies the types and formats of command and telemetry messages and the sequences of their interactions.
- SMCP assumes that any onboard component is designed according to FMS. Therefore, it is designed to monitor and control FOs.
- Since the characteristics of FOs can be retrieved from SIB, the design of SMCP is very simple.
- The formats of command and telemetry messages for each specific onboard component are also stored in SIB.

Advantages of This Approach

- The functional design of onboard components can be specified in a standard way, and reuse of functional design of onboard components will be easier. This reduces the cost of developing onboard components.
- The information stored in SIB can be used by any application program. This eliminates the cost and risk of manual information management among different teams using documents.
- Generic application programs that can be used for design, testing or operations of any component on any spacecraft can also be developed because these programs can retrieve information on specific components from SIB. This reduces the cost of developing supporting systems.
- Furthermore, part of the software installed in onboard components can be automatically generated from the information stored in SIB. This reduces the cost of developing onboard components further.

Elements of This Approach



Functional Model of Spacecraft (FMS)

Functional Model

- The Functional Model of Spacecraft (FMS) provides a framework for designing the functions of onboard components.
- FMS was developed based on
  - the Functional Viewpoint defined in the Reference Architecture for Space Data Systems (RASDS) developed by the Cousultative Comittee for Space Data Systems (CCSDS), and
  - the Computational Viewpoint defined in the Reference Model of Open Distributed Processing (RM-ODP) developed by ISO.
- But FMS adds many extensions to these Viewpoints.
- To design complex functions performed by onboard components, it is a common practice to decompose the functions into groups of functions, each consisting of a small set of functions closely inter-related to each other.
- In FMS, a group of functions defined in this way is called a Functional Object (FO).

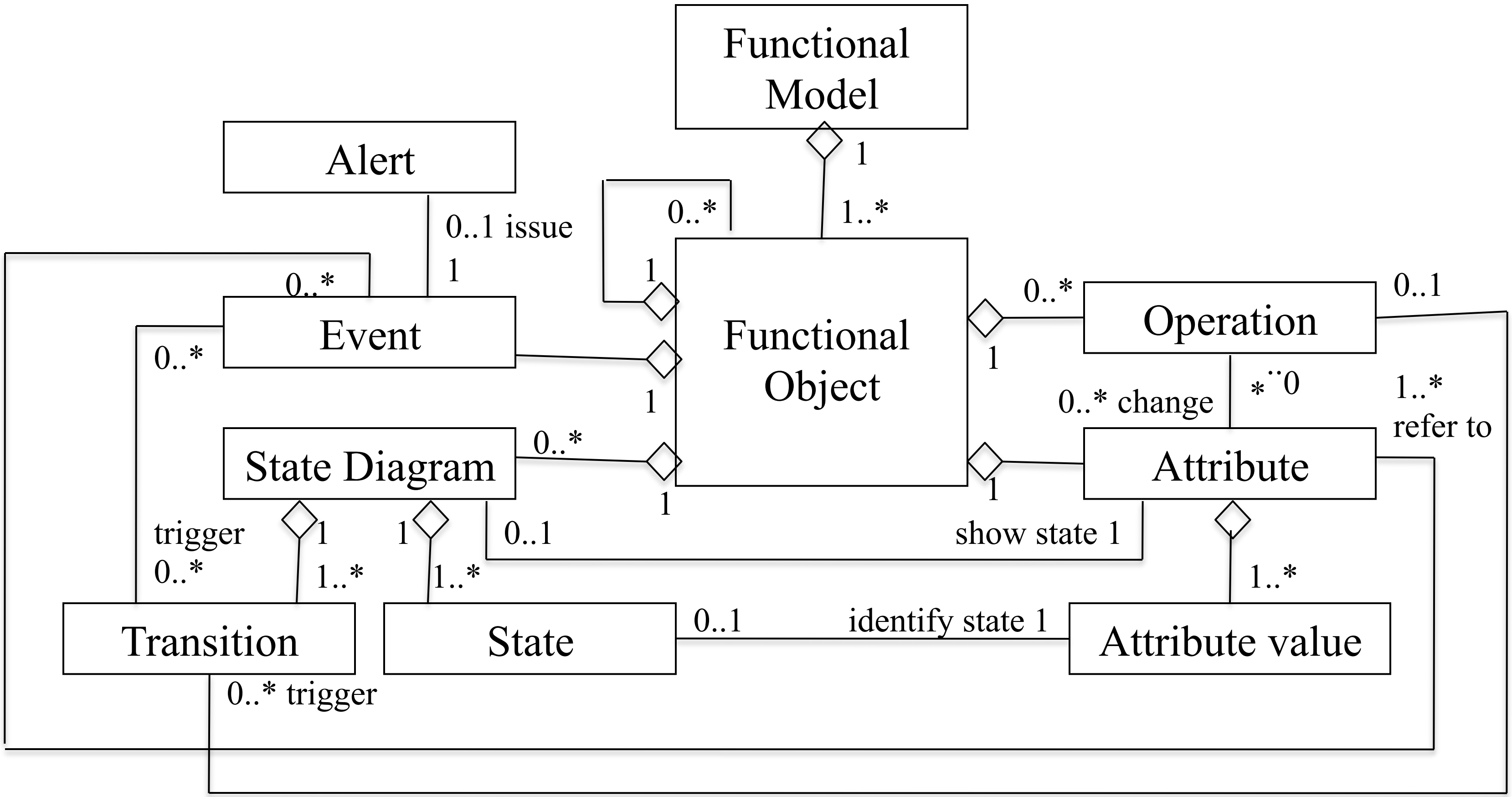
Functional Object

- A Functional Object (FO) can contain other FOs.
- A FO is specified with the following concepts:
  - Attributes,
  - Operations,
  - Alerts, and
  - Behavior.
- An attribute is a parameter that represents the status of a certain part or a certain aspect of the FO.
- An operation is a function performed by the FO and is invoked by receiving a command message from the outside of the FO.
  - Operations include those for setting values to attributes.
- An alert is a function to report to the outside of the FO that an important event has occurred inside the FO. Alerts are delivered to the outside world using telemetry messages.

Behavior of Functional Object

- The behavior of a FO is represented with one or more state diagrams.
- A state transition in a state diagram is triggered either by the execution of an operation or by the occurrence of an internal event.
- A state diagram is either active or inactive at any time.
  - Whether a particular state diagram is active or inactive is determined by what state the FO is in at that time in another active state diagram.
- What state of a state diagram the FO is in at a particular time is indicated by the value of an attribute associated with that diagram.
- Each state of a state diagram determines the set of operations that can be invoked when the FO is in that state.
  - Therefore, state diagrams specify the correct sequences of operations that can be performed by the FO.

UML Representation of FMS

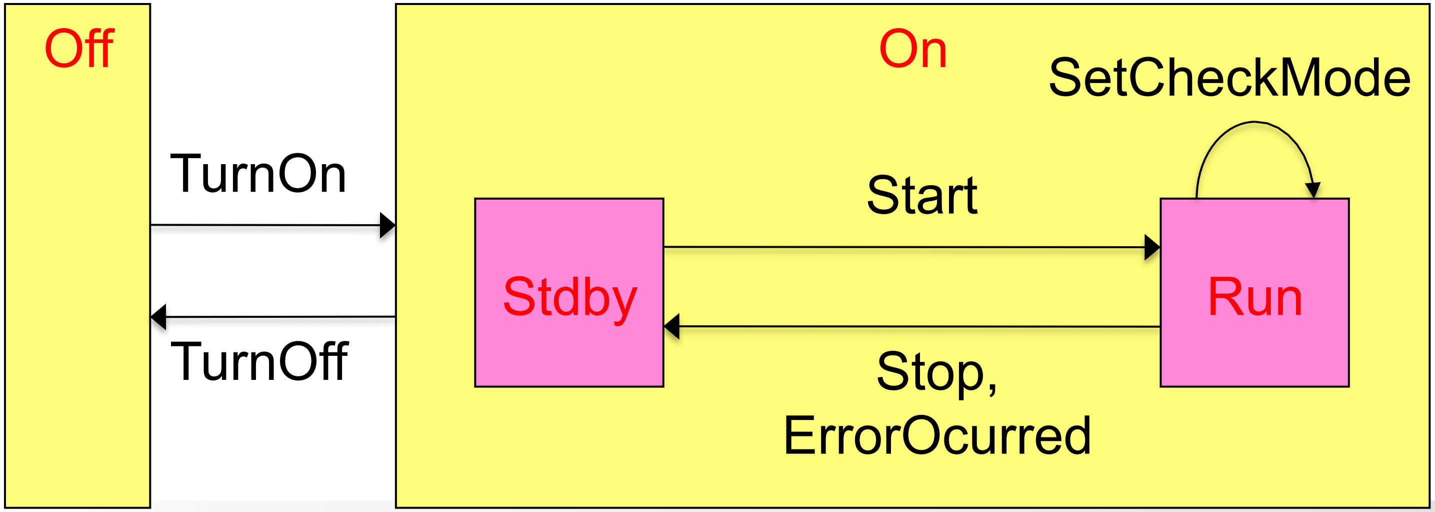


An Example of a Functional Object

- State Diagrams
  - OnOff (States: On and Off)
  - RunStandby (States: Run and Standby)
- Attributes
  - OnOff
  - RunStandby
  - ErrorStatus
  - CheckMode
    - The first two attributes show the states that this FO is in.
- Operations
  - TurnOn
  - TurnOff
  - Start
  - Stop
  - SetCheckMode
- Alert
  - ErrorDetected

Behavior of This Functional Object

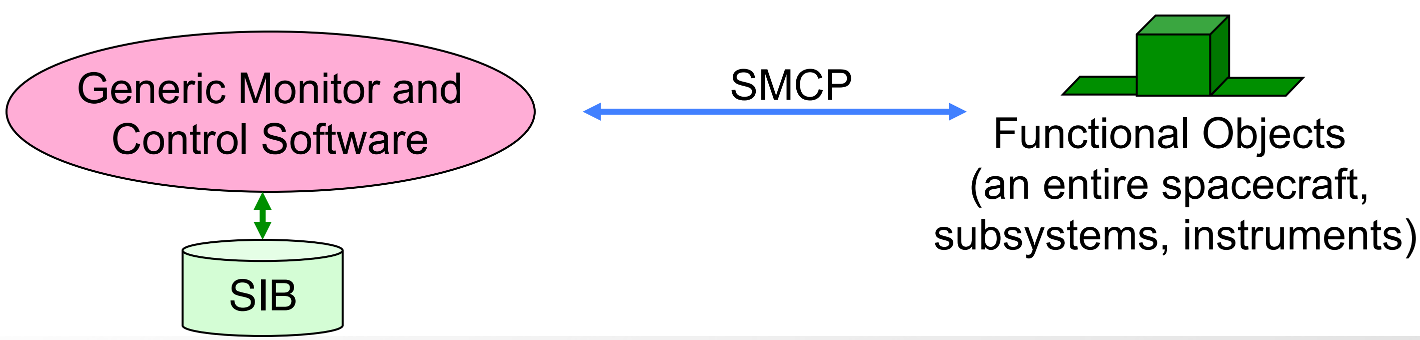
- The behavior of this FO can be represented with two state diagrams.
- Since the second diagram is active only when the FO is in state On of the first diagram, it is shown contained in the On state of the first diagram.
- The labels attached to the arrows in the diagrams are the triggers of the transitions.
  - ErrorOccurred corresponds to an internal event.
  - The other triggers correspond to the operations.



Spacecraft Monitor and Control Protocol (SMCP)

Spacecraft Monitor and Control Protocol (SMCP)

- To monitor and control onboard components, command messages are sent to the components and telemetry messages are generated by the components.
- The Spacecraft Monitor and Control Protocol (SMCP) is an application layer protocol that defines formats of command and telemetry messages used to monitor and control onboard components represented as FOs.
- SMCP can be use to
  - Control an entire spacecraft by the ground control system,
  - Control onboard subsystems by the central onboard computer,
  - Control an instrument by the instrument processor.



SMCP Messages

- Command message types
  - Action Command: Used to invoke an operation of a FO
  - Get Command: Used to trigger generation of a Value Telemetry message
- Telemetry message types
  - Value Telemetry: Used to send the values of an attribute or a set of attributes. Can be generated either
    - Periodically,
    - When there has been a change in the value of an attribute, or
    - When a Get Command is received
  - Notification Telemetry: Used to inform of an alert issued by a FO
  - Command Acknowledgement: Used to acknowledge receipt of a command message

Spacecraft Information Base (SIB)

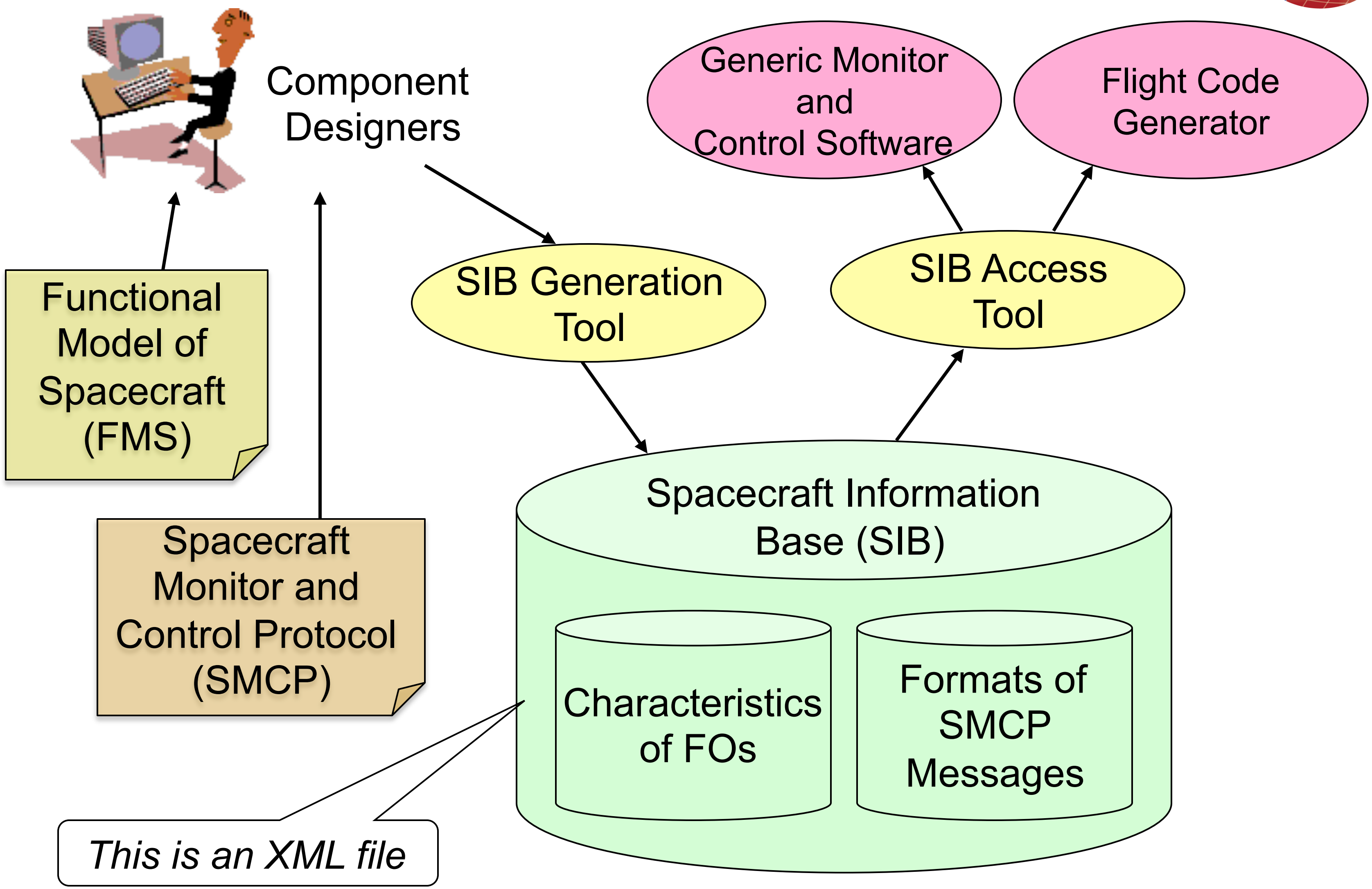
Spacecraft Information Base (SIB)

- The Spacecraft Information Base (SIB) is a standard database that stores:
  - The characteristics of the FOs developed for a spacecraft.
    - Attribute names, attribute data types;
    - Operation names, operation parameter names, operation parameter data types;
    - Alert names, alert parameter names, alert parameter data types; and
    - States, attribute values for each state, state transitions, trigger for each transition.
  - The formats of command and telemetry messages defined for a spacecraft according to SMCP
    - Telemetry message formats; and
    - Command message formats.

Generation of SIB

- The contents of SIB are generated by the designer of the onboard components and maintained throughout the lifetime of the spacecraft (that is, during design, component testing, spacecraft integration, flight operations, etc).
- We also developed an Excel-based tool to assist generation of the contents of SIB.
- The contents of SIB generated with this tool are automatically converted to an XML file, which can be used by any application program, including the code generator explained in the next section.
- We will develop another tool for generating the contents of SIB using graphical interfaces.

Generation and Utilization of SIB



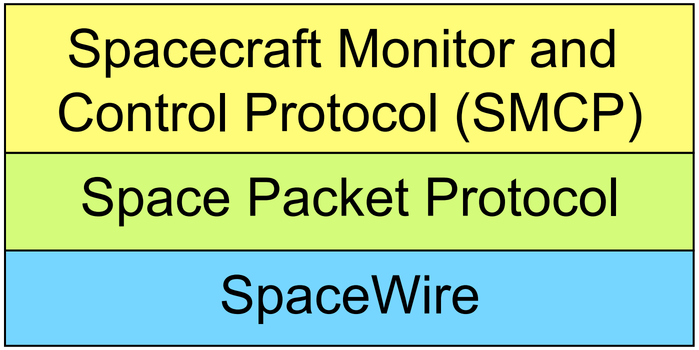
Excel-Based SIB Generation Tool

functionalObject definition										kind	kind property
0	1	2	3	4	5	6	7	8	9		
Nozomi										**functionalObject**	
	ScienceInstruments									**functionalObject**	
		PWA								**functionalObject**	
			CPU_Status							**stateAttribute**	**stateMachine** stateMachineCPU_Status
			CPU_RunStatus							**stateAttribute**	**steteMachine** stateMachineCPU_RunStatus
			HTR_Status							**attribute**	Enumeration {Ena, Dis}
			ECC_Status							**attribute**	Enumeration {Ena, Dis}
			ECC_1BitErrStatus							**attribute**	Enumeration {Error, NoError}
			ECC_2BitErrStatus							**attribute**	Enumeration {Error, NoError}
			WDT_Status							**attribute**	Enumeration {ENA, DIS}
			WDT_ErrStatus							**attribute**	Enumeration {Error, NoError}
			PWA_PacketFormat							**attribute**	Enumeration {1, 2, 3, 4}
			CPU_ON							**operation**	
			CPU_OFF							**operation**	
			CPU_RUN							**operation**	
			CPU_RST							**operation**	
			HTR_ENA							**operation**	**attributeChangeRule** changeRuleHTR_StatusEna
			HTR_DIS							**operation**	**attributeChangeRule** changeRuleHTR_StatusDis
			ECC_ENA							**operation**	**attributeChangeRule** changeRuleHTR_StatusEna

Development of Onboard Software

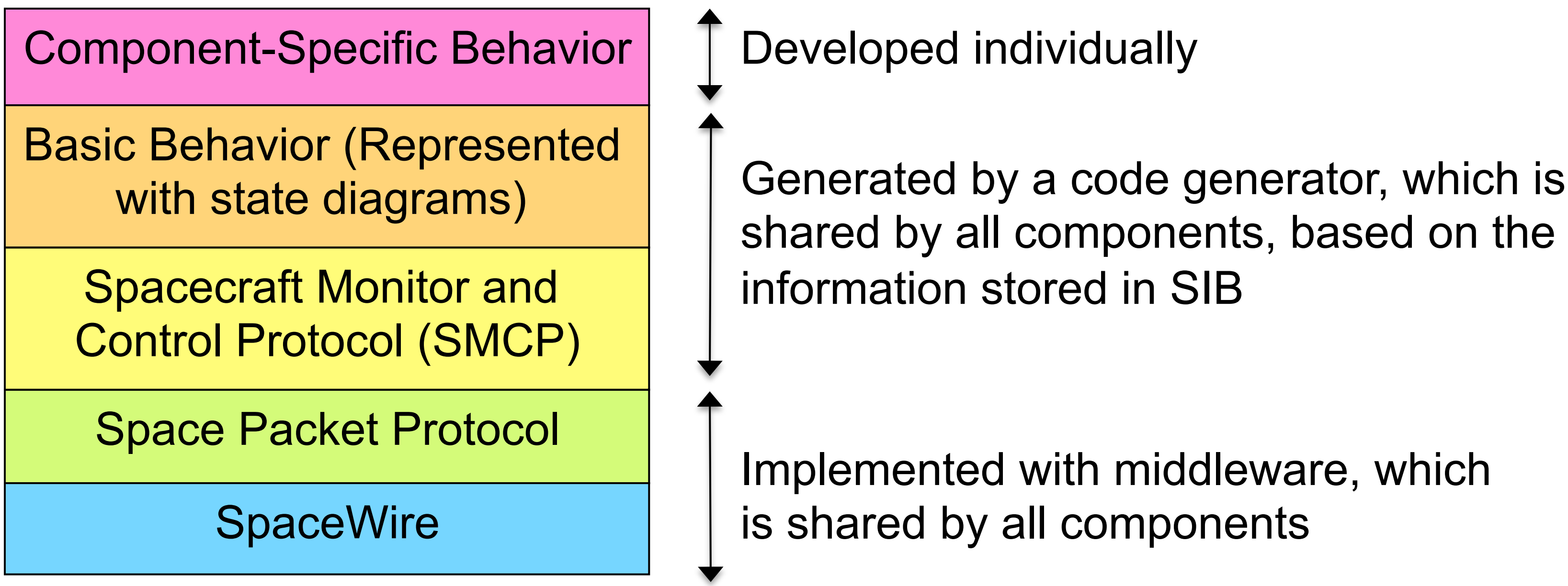
Standard Communications Protocols

- Japan Aerospace Exploration Agency (JAXA) is developing a standard onboard data handling architecture, which will be used for all future science spacecraft of JAXA.
- This architecture specifies, among other things, a stack of standard communications protocols to be used for communications between onboard components.



Development of Onboard Software

- Software used for onboard components will also be developed based on this architecture.



Conclusion

Conclusion

- In this paper, we have shown how the spacecraft onboard components can be developed based on the Functional Model of Spacecraft (FMS).
- The cost of developing spacecraft can be greatly reduced with this model-based method because:
  - Designers of onboard components can use the standard design and description methods, and do not have to develop or learn different design and description methods;
  - Reuse of designs of onboard components becomes easier;
  - Generic application programs that can be applied to any component can be developed by using the information on specific components stored in SIB;
  - Some part of flight software can be generated automatically by using the information stored in SIB; and
  - Manual exchanges of documents can be reduced by using SIB.

Applications

- JAXA is presently developing spacecraft onboard components based on the concepts pesented here for
  - SPRINT-A (an EUV telescope mission, launched in 2012), and
  - ASTRO-H (an X-ray telescope mission, launched in 2014).
- We will also apply these concepts to all future space science projects of JAXA.