Object-Oriented Model of Microsoft Solutions Framework

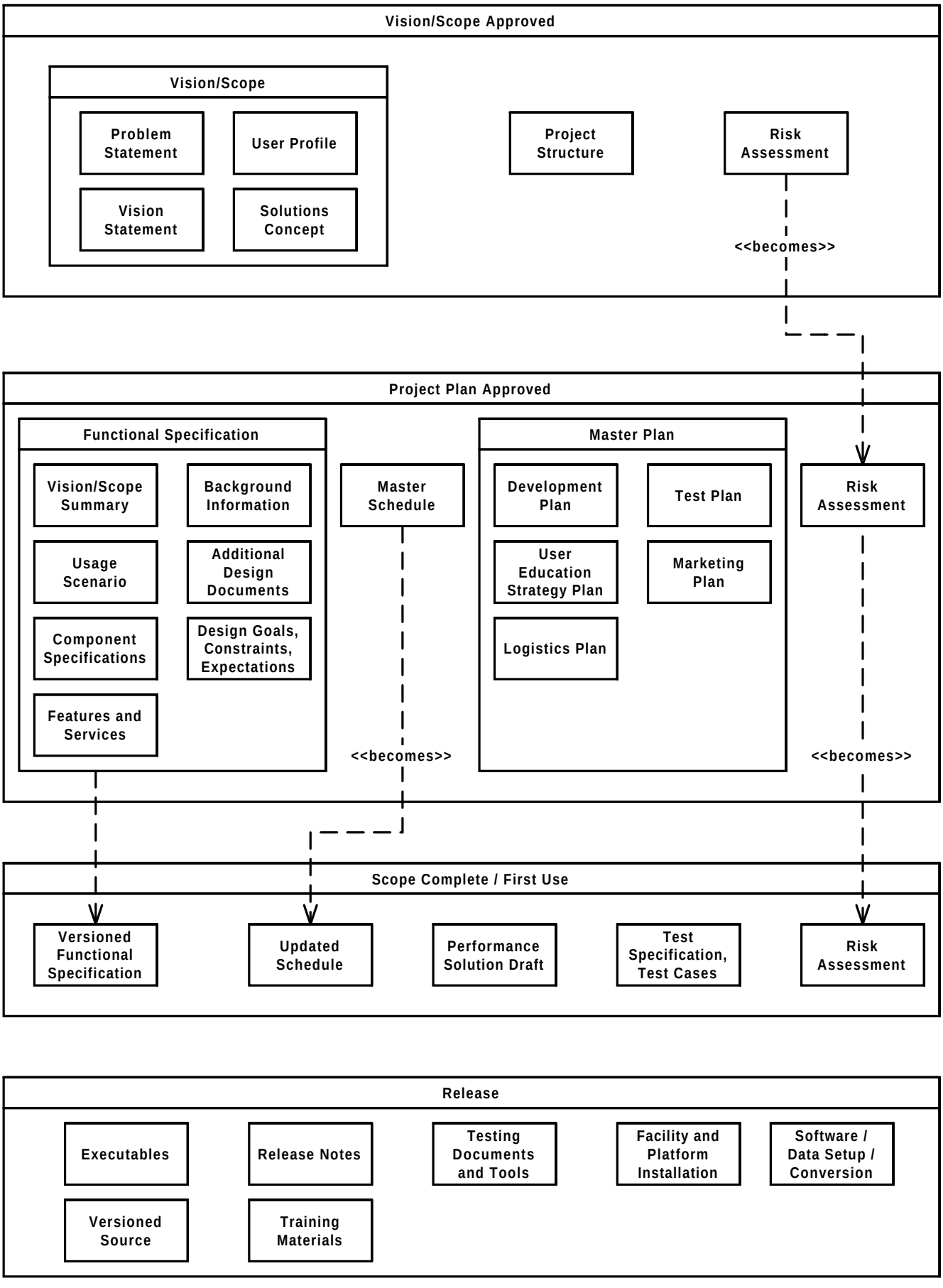
Characteristics of Original Process

The MSF (Microsoft Solutions Framework version 2.0, Microsoft, 1997) is a set of guidelines for developing client-server systems. MSF establishes a common framework that defines a team model, a process model and an application model. MSF can include user-defined processes (called best practices) that are compatible with the framework. The MSF process model is deliverable-based and can therefore quite naturally incorporate concrete processes defined in the object-oriented process model focused on here.

Benefits of Object-Oriented Representation

- Users can easily find the information they want that is otherwise non-systematically spread in various documents.
- An object-oriented model shows missing information that has to be specified by best practices.
- An object-oriented model provides a consistent framework for process extensions and definition of best practices.

Static Structure Diagram of Microsoft Solutions Framework



Examples of Process Class Specifications

(Please refer to the training material for illustrative examples of deliverable instances)

Problem Statement
«constructor» Review preexisting documents, conduct interviews, ask questions «quality assurance» stay at very high level, do not get too technical
Purpose (responsibility): Establish what is motivating the initiation of the project Owner: Product Management Standard: See MSF training material

Vision / Scope Approved
Vision / scope Risk assessment Project structure
«constructor» see deliverables: Vision/Scope, Risk assessment, Project structure «quality assurance» see deliverables: Vision/Scope, Risk assessment, Project structure
Purpose (responsibility): opportunity for the team to agree upon the product direction and what will an will not be in the product

Vision Statement
«constructor» Review existing material, conduct interviews «quality assurance» Balance all the interests to arrive at a single vision statement, surface enterprise architecture implications early. Be specific, measurable, achievable, realistic, time-ordered
Purpose (responsibility): establish long term vision, provide content for design making Owner: Product Management Standard: See MSF training material

User Profile
Type of user Physical location Language Educational background Job function Job tools and reference materials User population count Access and security needs Hardware and software configuration Special usability requirements.
«constructor» Conduct interviews, build customer profile matrix, prioritize conflicting goals, gather acceptance criteria «quality assurance» Do not over-constrain the problem, must keep a management perspective in making trade-offs
Purpose (responsibility):to identify the customers so as to gain an understanding on managing goals , expectations Owner: Product Management Standard: See MSF training material

Solutions Concept
Project success factors (how success will be defined and measured) Operations concept (establish and understand the workflow) Deliverables (list of components which will make the new product operational) Acceptance criteria (checklist of requirements that must be satisfied before the product goes to production).
«constructor» Review existing material, build on the vision statement, keep in mind the use of this document «quality assurance» not defined
Purpose (responsibility): outline an approach that will provide the basis for planning and scoping for the next milestone Owner: Program Management Standard: not defined

Risk Assessment

Risk mitigation matrix (severity, probability, impact, description, mitigation plan)

«constructor»
Succinctly identity the key risk
Outline a risk mitigation plan
Flag each risk by red, yellow and greens color

«quality assurance»
Risks are reviewed in status meetings at milestones

Purpose (responsibility): identifies the upcoming technologies that should be monitored, identifies the organizational issued that might impact the process of the project
Owner: team leads
Standard: See MSF training material

Project Structure

E-mail name
Telephone list
Mail address
Server share name' directory structure
Other information critical to team organization.

«constructor»
Update the document at the Project Plan Approved milestone when project resources are assigned
Publish the document, for example, on internal web-page

«quality assurance»
this is preliminary document

Purpose (responsibility): defines administrative structure for the project team
Owner: program management
Standard: not defined

Functional Specification

Vision scope summary
Background information
Design goals, usability goals, constraints, expectations
Usage scenarios
Features and services
Component specifications
Additional conceptual design, logical design, and physical design

«constructor»
Address what needs to be included in the product
Assure completeness: provide formal inspections, design state tables, apply usage scenarios, write example test cases to determine whether the specification is sufficiently detailed to test against.
Review and negotiate functional specification
Approve functional specification

«quality assurance»
The team has enough information to commit to a release date
The customer and team agree that the specification describes what needs to be delivered.

Purpose (responsibility): Contract between the customer and the product team and is the basis for building a project plan and schedule. Detailed description of the product or service.
Owner: program management
Standard: not defined

Master Plan

Development plan
Test plan
User education strategy plan
Logistics plan
Marketing plan

«constructor»
not defined

«quality assurance»
not defined

Purpose (responsibility): Specify the tasks required to complete the project, the dependencies between these tasks and the order in which the tasks are to be performed.
Owner: program management
Standard: not defined

Master Schedule
Development schedule Test schedule User education schedule Logistics schedule Marketing schedule
«constructor» Developers estimate their tasks. All other roles base their schedules on development schedule. Controversial issues are put into buffer to allow for unknown tasks. «quality assurance» Schedule is risk-based, includes external constraints and events.
Purpose (responsibility): estimating tasks on critical path. Owner: all team leaders Standard: not defined

Scope Complete /First Use
Versioned functional specification Updated schedule Performance solution draft Test specification and test cases Risk assessment
«constructor» Develop code Test usability Implement hardware, software and network Asses risks and mitigation Review code «quality assurance» All the features of the product work Testing may not be complete, but the product can be used and evaluated. Target customers can use the product for the fist time under beta test conditions.
Purpose (responsibility): to develop stable key deliverables

Versioned Functional Specification
Refer to the MSF document “What to include in your functional specification”
«constructor» Refer to the MSF document “Functional Specification Guidelines” «quality assurance» Refer to the document “Functional specification checklist”
Purpose (responsibility): to define the product exactly Owner: Program Management Standard: not defined

Release
Executables Release notes Training manuals and user performance solutions Testing documents and tools Facility and platform installation Software / data setup / conversion
«constructor» Stabilize performance support and training Beta releases Prepare sites Obtain customer sign off Roll out Post-implementation review «quality assurance» Refer to the MSF document “Software release deliverable checklist”
Purpose (responsibility): the product is released to operation and support groups

Training Materials
«constructor» not specified «quality assurance» not specified
Purpose (responsibility): not specified Owner: user education Standard: not defined