ENTERPRISE ARCHITECTURAL SPACE ORGANIZING TABLE

			1	2	3	4	5	6	7
		Viewpoints	Purpose (Why)	Data (What)	Function (How)	Timing (When)	Network (Where)	People (Who)	Scorecard (Test)
Business Architecture	A	CEO	- Strategic scope - Economic intent	Enterprise level dashboard	Enterprise process models	Business events and cycles	Locations where the enterprise operates	Organizational units	Balanced scorecard EVA, ROE, ROA
	B	General Mgr	Operational goals and objectives	Operational dashboard	Business unit process models	Business unit master schedule	Business unit logistic network (nodes and links)	Business unit org chart with roles, security permissions and skill sets	P&L DSO
	C	Process Owner	Process control objectives	Process control dashboard	Application process models	Process schedule, events, collaborations and state transitions	Communication links and devices needed for process automation	Process level actors with roles and permissions	%Utilization Defect Rates Cycle Times Step Count
	D	Process Worker	-Activity objectives	Specific data inputs, processing and outputs	Task and activity based process models	Activity specific events , collaborations and state transitions	Activitiy specific devices and locations	Specific roles, accounts, passwords and permissions	Activity metrics
Integration Architecture	E	Enterprise Architect	- Extended enterprise use cases - EAI use cases	Enterprise level data flows and replication strategies	EAI (API, method, data, user interface) - B2B integration -Global (GXA) - Message broker	Define business process models -BPEL4WS -Rosetta PIPs	-URI's for all trading partners -Integration servers and firewalls	Partners. customers, suppliers, system actors	- EE acceptance test - Performance against SLAs - Onboarding costs
	F	Designer	System level use cases	XML Schemas, Half Maps,ETL, Batch feeds	Design level service, component, and subsystem models	Design level collaboration models	- Integration NFR* met	Roles, permissions, security requirements	EE integration tests - Reuse - Transaction metrics
	G	Developer	Abbreviated use case descriptions Implement interfaces	Profiles, database instances, stored procs	XLANG, source code, scripts, batch files, executables	XLANG Executable vertical and horizontal slices	Process mapped to processors, links, protocols	Users mapped to roles within organizations	Unit tests, system tests
Application Architecture	H	Enterprise Architect	Enterprise level use cases Process refactorings NFR*	- Enterprise data models - Data distribution strategies	Enterprise level domain model and logical services	Enterprise level collaboration models (sequence and interaction)	Enterprise level system architecture -Nodes -Links -Locations	Enterprise level user profiles including demographics, psychographics, technographics	-H1 realizations -Cross application integration test -SLA metrics -Reuse
	I	Architect	Application level use cases Mechanisms NFR	Application level data models	Application level -Domain models -Analysis model	Analysis level collaboration model	Application level system archtecture - Nodes, devices - Links and segments - Processors	Information architecture, interaction maps, story boards, security requirements	I1 realizations Acceptance tests App SLA metrics Reuse
	J	Designer	System level use cases Mechanisms NFR*	Tables, indexes, views, queries	Design level classes, component, subsystem and QoS	Design level collaboration models (sequence and interaction)	Design level system architecture - addresses -subnets -processors	Visual designs, wire frames, site maps	J1 realizations Integration tests Interfaces defined Reuse
	K	Developer	- Abbreviated use case descriptions - Implement interfaces	Database instances , stored procs, etc.	Source code Implemetation units, Executables	Executable (vertical / horizontal slices)	-Processes allocated to processors - Production environ - NFR* met	Intuitive, easy to use executable interface relevant to user needs	Unit Tests User Adoption
Operational Architecture	L	Systems Architect	Non functional requirements - high level	-Directory Design -Data storage design	-Monitoring and tuning -Remote management	Event management	High level network models Traffic analysis	Users, roles, permissions, security requirements	-Performance against SLA's - Operational metrics
	M	System Engineer	Non functional requirements - detailed level	- Directory Implementation - Backup and recovery	Batch files, scripts, utilities	Fault management and recovery	- Detailed network models - Network monitoring	User administration	Operational metrics
Development Architecture	N	Config Mgmt Engineer	- Change impact analysis - Rollbacks - Asset retention	-Repository -Dependency maps	Batch files, scripts, utilities	-Restore known configurations -Promote code	Logical and physical device information	Users, roles, permissions, security requirements	Asset management metrics
	O	Buildmaster	-Build quality and quantity metrics -Reports	Source code, compile time dependencies, test data, results,	Compilers, build tools, system admin tools, test tools	Development process, events and schedule	Development, test, and staging environment locations and accounts	Development team, system admin team, configuration mgmt team	-Build quality and quantity metrics -Reports
	P	Test Engineer	-Quality and predictability metrics -Reports	-Test Cases -Test Data -Repository	Automated test suites - integration - acceptance - performance	Test plan with schedule and test cases	Test environment	Development team, buildmaster, Config mgmt team, QA mgmt	-Quality and predictability metrics -Reports
	Q	Developer	-Software development efficiency and effectiveness -Reports	- Repository - Test Data	- IDE - Bug Tracking Tools - Debugging Tools - Test Tools - Modeling tools	-Iteration plan with schedule and features - Integration schedule	Development environment including integration machine(s)	Users, roles and permissions	-Software development efficiency and effectiveness -Reports

* Non-Functional Requirements