



# **Collaborative Scheduling Using JMS in a Mixed Java and .NET Environment**

Yeou-Fang Wang, Ph.D.

Allan Wax

Ray Lam

John Baldwin

Chet Borden

**JPL/CalTech**



# Overview

- A proof-of-concept prototype successfully demonstrated a collaborative scheduling solution for future NSAS Deep Space Network application
- A prototype distributed computing environment was established for collaborative scheduling
- Java Message Service (JMS) was used in a mixed Java and .NET environment for messaging

# Agenda

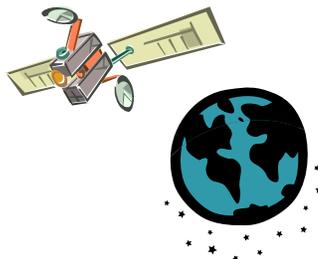
- NASA Deep Space Network scheduling
- Collaborative scheduling concept
- Distributed computing environment
- Platform concerns in a distributed environment
- Messaging and data synchronization
- The prototype
- Conclusion



# NASA Deep Space Network (DSN)



Goldstone, CA  
Canberra, Australia  
Madrid, Spain



30+ missions

**Scheduling** is to arrange antenna times for space flight missions and ground-based science observations (requirements) under certain restrictions (constraints)

Requirement: N tracks every M days, Complex pattern-based coverage, Antenna arraying, Multiple Spacecraft Per Antenna, ...

Constraint: Ground assets: Antenna, Equipment, Downtimes, ...  
Mission: Viewperiod, Horizon mask, Setup/teardown, ...  
Combined: BOT, RFI, ...



# DSN Scheduling Users

- DSN schedulers:
  - Schedule owner
  - Prepare maintenance schedule
  - Coordination
- Space flight project schedulers:
  - Provide requirements
  - Negotiation and proposals
  - Submit changes
- Station operators:
  - Carry out schedule
  - Handle contingency and anomalies



# Current Scheduling System

- Mainly a manual process with software support
- Long-range, mid-range, near-real-time processes handled by different groups
- Various tools are deployed for each process
- Meetings to resolve conflicts



# Future Scheduling System

- Seamless scheduling for all planning horizons
- A master schedule always exists, visible to all users
- Requirements and schedules are fully traceable
- Conflicts are resolved at the lowest level possible in a peer-to-peer fashion
- Meetings are called only as needed
- Workspace is provided to users to develop requirements and for what-if analysis
- Distinguish global (shared) workspace and local (private) workspace
- Private workspace may span a set of peers
- Need scalability (loading, # users, # assets) and extensibility (evolving technology)
- Intelligent assistants for decision support

# Collaborative Scheduling Concept

- Decision Makers (DM): DSN schedulers, project schedulers, managers, operators, ...
- Every one knows their specific requirements and constraints
- No single DM makes decisions for others (distributed decision making)
- DMs share information
- Shared responsibility to create a successful schedule
- Work as a team to resolve conflicts



# Collaborative Scheduling Prototype Features

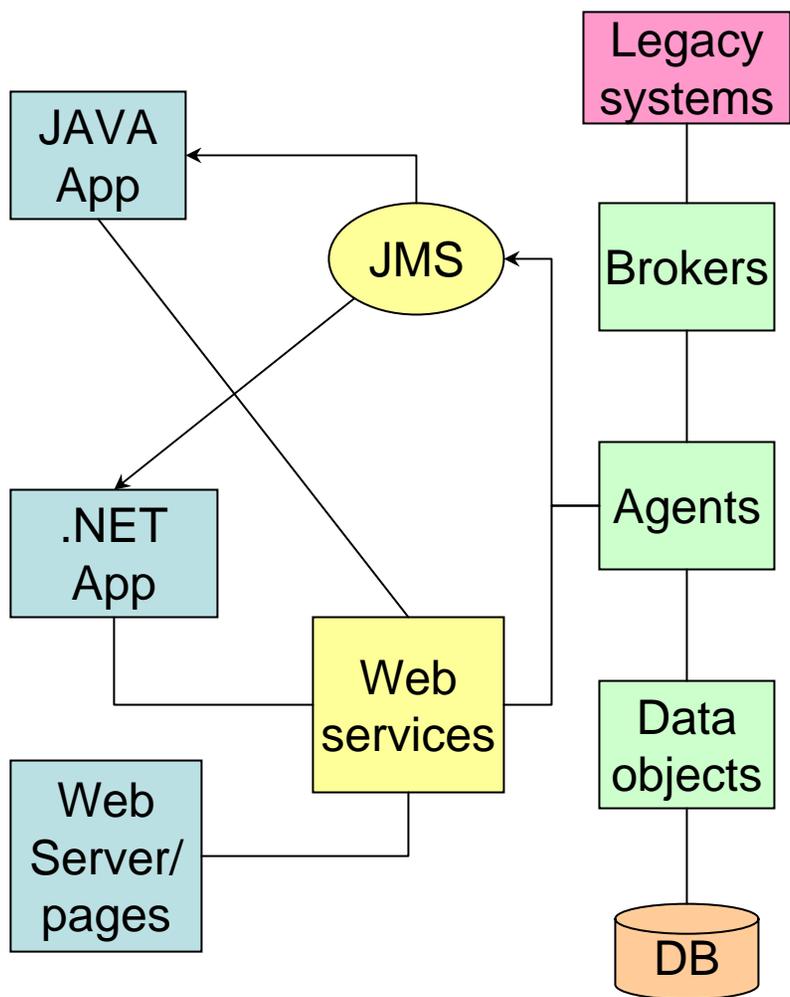
- Single master schedule shared by all users
- Conflict-aware scheduling
- Dynamic workspace/static workspace
- Private workspace with limited sharing
- Scenario management for comparison
- Synchronous/Asynchronous collaboration (e.g. conflict resolution, negotiation)
- Ownership and workflow management
- Data synchronization
- Messaging/notification/alert
- Intelligent assistant
- Traceability
- Distributed computing environment



# Distributed Computing Environment

- **Objective:** To connect users and resources in a transparent, open, and scalable manner.
  - ***Transparency:*** distributed experts are working together as if they are co-located
  - ***Openness:*** provides each project with a continually open environment that enables interaction with other projects until a satisfactory condition exists.
  - ***Scalability:*** The solution should be able to accommodate changes in the number of user projects and ground resources in the DSN domain.

# Architecture



- Data
- Data access
- Legacy information
- Business logic
- Web services
- Messaging
- Client interfaces



# Platform concerns

## in a distributed environment

- Distributed computing environment should be open and flexible for multiple frameworks.
- JAVA and .NET are two major players now:
  - It is expected that they will reach a 50-50 market share by the end of this year (based on articles from Gartner, Meta group, ZDNet, ...). Each will probably share 30%-40% market.
  - Java: large-scale enterprise, multi-platform
  - .NET: small/mid-size development, easy to use, performance/speed advantage
- There may be other frameworks emerging in the future
- Take advantage of each framework based on our needs



# Possible Cross-Platform Solutions

- Java: through virtual machines for many platforms
- .NET and Mono: provides the necessary software to develop and run .NET client and server applications on Linux, Solaris, Mac OS X, Windows, and Unix.
- Mainsoft: .NET to J2EE
- Others such as Qt (Trolltech)



# Messaging & Data Synchronization

- Collaboration requires instant communication
- All users are notified in real-time regarding data changes
- JMS is used for centralized messaging
- Time synchronization is the base for data synchronization
- Keep messages in messaging bus and keep data in data bus



# Prototype Implementation

- Use SOAP/XML-based Web services
- Agents and backend are implemented using .NET
- Web services are in ASP.NET
- Clients are in Java and .NET
- JMS for messaging
- IKVM is used for .NET to communicate with JMS

# IKVM

- IKVM.NET is an implementation of Java for Mono and the Microsoft .NET Framework.
- It includes
  - a Java Virtual Machine implemented in .NET
  - a .NET implementation of the Java class libraries and
  - tools that enable Java and .NET interoperability.
- Using IKVM, we can take advantages from both Java and .NET

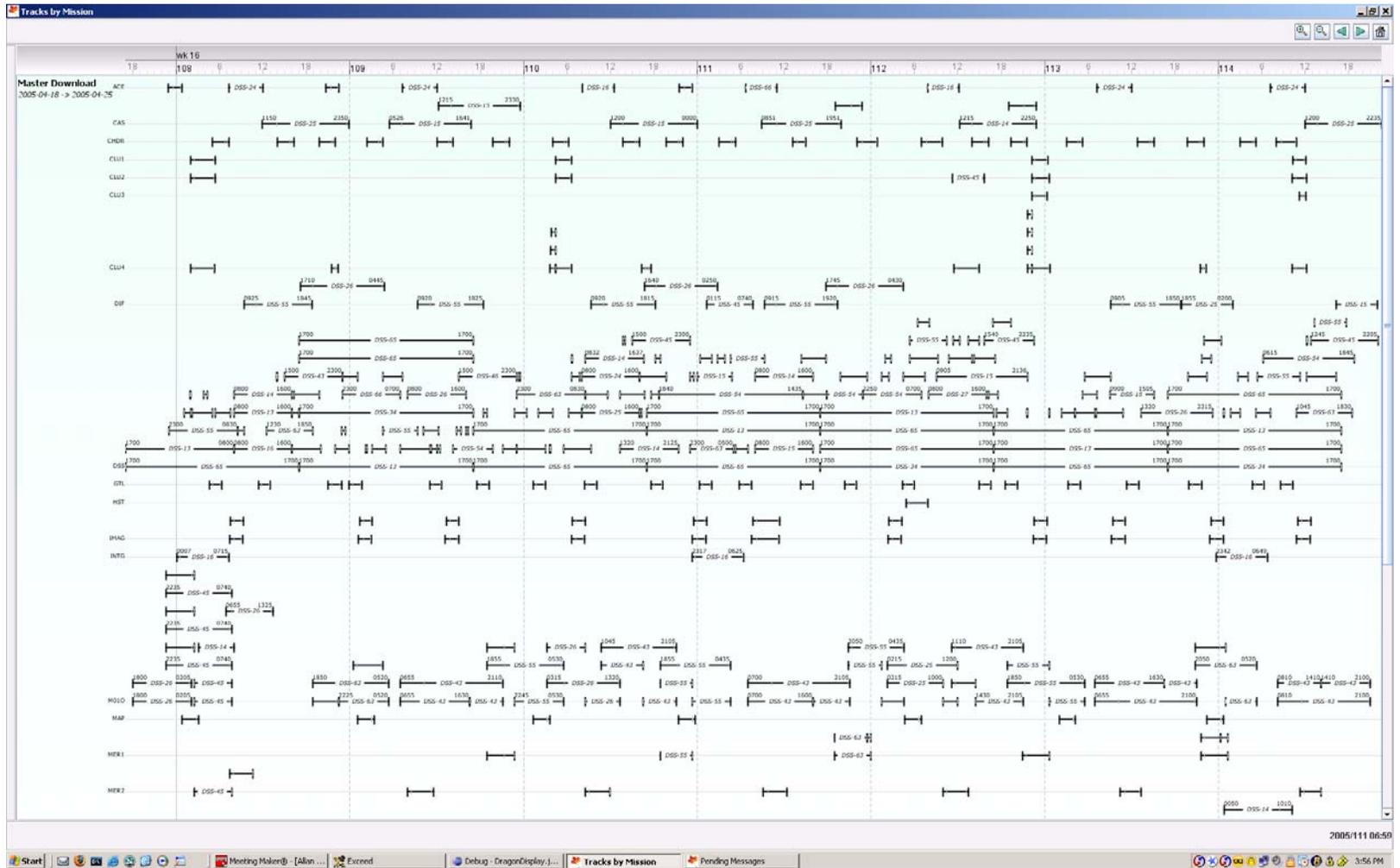


# The Prototype

- Database
  - Master schedule
  - Dynamic workspace/static workspace
  - Time-based ownership
  - Traceability information
- Middle-tier
  - Conflict-aware scheduling
  - Private workspace sharing
  - Scenario management for comparison
  - Synchronous/Asynchronous collaboration (e.g. conflict resolution & negotiation)
  - Workflow management
  - Data synchronization
  - Notification/alert
- Web services wrapper
- Clients
  - Java client for schedule viewing with dynamic update under user's control
  - .NET Integrated Analysis Environment for complete analysis experience
  - Web pages to view schedule



# Java Client





# .NET Integrated Analysis Environment

Crane/Master -- ProjSched2(01/01/00 00:00 - 12/31/20 00:00.ULYS)

File Edit View Scenario Tools Help

Shortcuts Schedule Views Forecast Views Supportability

Garnt Planner Supportability Compare

9/5/2005 (249) 9/6/2005 (249) 9/7/2005 (250) 9/8/2005 (251) 9/9/2005 (252) 9/10/2005 (253) 9/11/2005 (254)

Scenario

- Scenario
- Comm Points
- Missions
- Days

MonthView Calendar DayView Scenario

Details

Assets

ConsumerName	CAS
ResourceName	25
TargetName	CAS

Information

ENG NONE

Operational

CommitmentDueDate

ConcreteDate 12/31/9999 11:59 PM

Conditions (Collection)

Req\_Op None

Period

End	9/7/2005 11:59 AM
Setup	000
SDE	
Start	9/7/2005 7:50 AM
Teardown	000

Dynamic ID	Consumer	Target	Resource	State	Description	DOY	Start	End	Setup	Teardown	Conflicts
40	POLR	POLR	45	Release	PB ONLY/DL	250	09/07/05 02:30	09/07/05 04:15	030	015	
169	CHDR	CHDR	54	Release	TKG PASS	250	09/07/05 02:45	09/07/05 05:00	100	015	
461	DSS	DSS	26	Release	TDN PROF	250	09/07/05 03:15	09/07/05 05:40	000	000	
171	VGR1	VGR1	15	Release	TKG PASS	250	09/07/05 03:15	09/07/05 07:00	030	015	
281	SOHO	SOHO	34	Release	LGA MR PASS	250	09/07/05 03:45	09/07/05 07:50	045	015	
234	VGR2	VGR2	45	Release	TPX-HIGH PWR	250	09/07/05 04:15	09/07/05 08:25	030	015	
129	DSN	DSN	54	Release	ST35 MSTA	250	09/07/05 05:00	09/07/05 08:00	045	015	
268	GTL	GTL	66	Release	TR DUMP 131S	250	09/07/05 05:35	09/07/05 07:15	025	010	
285	MRO	MRO	26	Release	T/P CONTINUOUS	250	09/07/05 05:40	09/07/05 15:25	100	015	
26	DSS	DSS	65	Release	STA DIRECTED ACT	250	09/07/05 06:00	09/07/05 06:25	000	000	
496	DSS	DSS	46	Release	ANTENNA PHASING	250	09/07/05 06:00	09/07/05 10:00	000	000	
279	DSS	DSS	63	Release	STA DIRECTED ACT	250	09/07/05 06:25	09/07/05 08:15	000	000	
387	DSS	DSS	65	Release	STA DIRECTED ACT	250	09/07/05 06:25	09/07/05 07:10	000	000	
361	DSS	DSS	15	Release	TDN PROF	250	09/07/05 07:00	09/07/05 08:15	000	000	
305	SOHO	SOHO	65	Release	LGA MR D/L PASS	250	09/07/05 07:10	09/07/05 12:55	030	015	
233	SOHO	SOHO	66	Release	LGA MR U/L PASS	250	09/07/05 07:15	09/07/05 12:50	025	010	
9	DSN	DSN	55	Release	X/KKA SYS GAIN	250	09/07/05 07:15	09/07/05 18:10	000	000	
284	CAS	CAS	25	Release		250	09/07/05 07:50	09/07/05 11:50	000	000	
365	DSS	DSS	34	Release	STA DIRECTED ACT	250	09/07/05 07:50	09/07/05 08:20	000	000	

Console

```

Refreshing children: Start.List.Gannt.Planner.Forecast.Tree.End
Changes downloaded: (local 1/5/2006 2:54:52 PM, last update 1/5/2006 10:48:44 PM) D(113986a85-2382-4fec-bf3d-72f217738e7a) N(c19698cd-cb63-451e-9d1c-a0ef722d4449)
Changes uploaded: (local 1/5/2006 2:54:54 PM)
JMS message: CRANE*Scenario change*Master*09/05/05 00:00 - 09/12/05 00:00*D(c19698cd-cb63-451e-9d1c-a0ef722d4449) *Draft*ProjSched2*ULYS
Changes downloaded: (local 1/5/2006 2:54:55 PM, last update 1/5/2006 10:48:44 PM) D(113986cd-cb63-451e-9d1c-a0ef722d4449) D(113986a85-2382-4fec-bf3d-72f217738e7a)
Local scenario updated: (local 1/5/2006 2:54:55 PM) : D(c19698cd-cb63-451e-9d1c-a0ef722d4449)
Refreshing children: Start.List.Gannt.Planner.Forecast.Tree.End
  
```

Ready



# Web Page

CRANE Schedule View

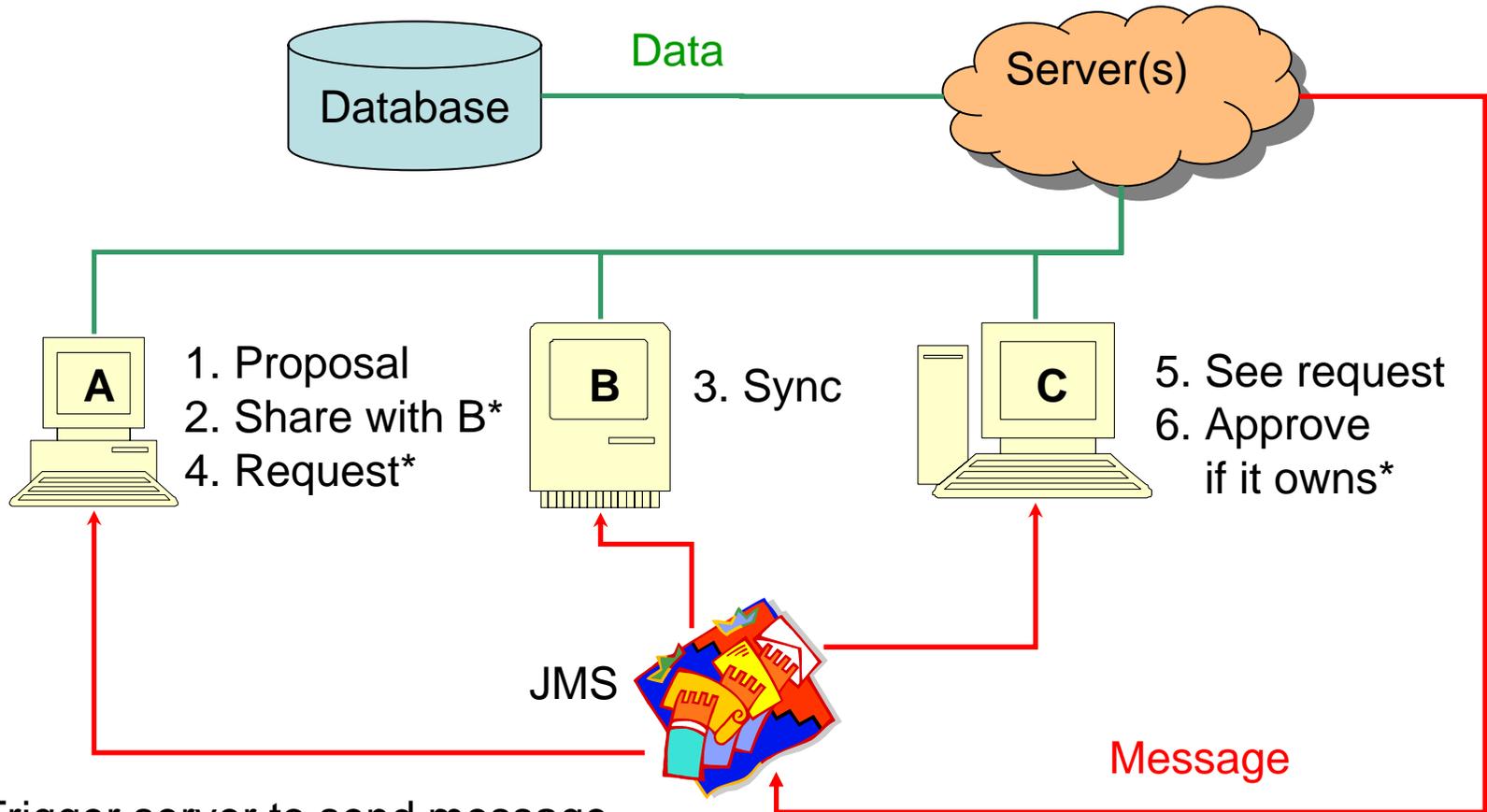
http://localhst/craneweb/ScheduleView.aspx

Scenario: Master 3/13/2006 12:00:00 AM - 3/20/2006 12:00:00 AM at time point 1/1/2007 12:00:00 AM  
Last Updated: 3/12/2006 2:49:40 PM

Lowest State: Release Interval: [INTERVAL] Mission: [ALL] Antenna: [ALL] Refresh New Interval Log out

LINE	START	SOA	EOA	FAC	USER	ACTIVITY	PRE	BOT	EOT	POST	WORK	CONFIG	XFLCT
ITEM	DOY			ID	NAME	DESCRIPTION	CAT.			CAT.	CODE	CODE	
<a href="#">Edit</a> 1	72	0000	1200	16	DSS	STATION CLOSED	000	0000	1200	000	2B4	NONE	
<a href="#">Edit</a> 2	72	0000	1600	13	DSS	STA DIRECTED ACT	000	0000	1600	000	2B2	NONE	
<a href="#">Edit</a> 3	72	0025	0245	54	DSS	STA DIRECTED ACT	000	0025	0245	000	2B2	NONE	
<a href="#">Edit</a> 4	72	0025	0355	65	DSS	STA DIRECTED ACT	000	0025	0355	000	2B2	NONE	
<a href="#">Edit</a> 5	72	0055	0355	10	DSS	PREPAS VLBI TAPE	000	0055	0355	000	1A4	NONE	
<a href="#">Edit</a> 6	72	0055	0355	60	DSS	PREPAS VLBI TAPE	000	0055	0355	000	1A4	NONE	
<a href="#">Edit</a> 7	72	0140	0320	46	GTL	TR DUMP 131S	025	0205	0310	010	3C1	NONE	
<a href="#">Edit</a> 8	72	0205	0825	45	MGS	MAX3	115	0320	0825	000	1A1	N711	
<a href="#">Edit</a> 9	72	0205	1140	45	M010	MA11 0825DN012	115	0320	1125	015	1A1	N006	
<a href="#">Edit</a> 10	72	0245	0500	54	CHDR	TKG PASS	100	0345	0445	015	1A1	N083	
<a href="#">Edit</a> 11	72	0315	0600	43	SOHO	LGA HR D/L PASS	030	0345	0545	015	1A1	N073	
<a href="#">Edit</a> 12	72	0320	0555	46	SOHO	LGA HR U/L PASS	025	0345	0545	010	3C1	NONE	
<a href="#">Edit</a> 13	72	0355	0955	15	DSN	RFC CLK SYN D-M4	130	0525	0925	030	1A4	F302	
<a href="#">Edit</a> 14	72	0355	0955	65	DSN	RFC CLK SYN D-M4	130	0525	0925	030	1A4	F303	
<a href="#">Edit</a> 15	72	0400	0800	66	DSN	PERF EVAL TEST	030	0430	0730	030	2A5	NONE	
<a href="#">Edit</a> 16	72	0435	0700	55	DSS	STA DIRECTED ACT	000	0435	0700	000	2B2	NONE	
<a href="#">Edit</a> 17	72	0435	0705	14	DSS	TDN PROF	000	0435	0705	000	2C2	NONE	
<a href="#">Edit</a> 18	72	0435	0745	26	DSS	TDN PROF	000	0435	0745	000	2C2	NONE	
<a href="#">Edit</a> 19	72	0435	0755	63	VGR1	TKG PASS	030	0505	0740	015	1A1	N007	
<a href="#">Edit</a> 20	72	0500	0650	54	DSS	STA DIRECTED ACT	000	0500	0650	000	2B2	NONE	
<a href="#">Edit</a> 21	72	0500	1235	34	DSS	MSPA TDN DEV	000	0500	1235	000	2C2	NONE	
<a href="#">Edit</a> 22	72	0555	0830	46	POLR	PB OPS	025	0620	0820	010	3C1	NONE	
<a href="#">Edit</a> 23	72	0600	1130	43	MRO	T/P CONTINUOUS	100	0700	1115	015	1A1	N002	
<a href="#">Edit</a> 24	72	0650	1005	54	WIND	TR DUMP	100	0750	0950	015	1A1	N083	
<a href="#">Edit</a> 25	72	0700	1515	55	DSS	MAINTENANCE	000	0700	1515	000	2A1	NONE	

# How this works



Proposed change with ownership and no conflict: automatically accepted and sync  
Proposed change without ownership: workflow & ownership determine approval process

# Conclusion

- We have prototyped a collaborative environment for DSN scheduling and successfully proved the concept
- DSN scheduling is a system that involves multiple agents
- Collaborative scheme needs to be developed in a multi-agent environment
- Efficient communication and data synchronization is a key for collaboration