PROJECT CONTROLS SUBGROUP

MASTER PROGRAM SCHEDULE INTEGRATION METHODOLOGY

WHITE PAPER

FY 2024

Prepared by:
Energy Facility Contractors Group (EFCOG)
Project Delivery Working Group
Project Controls Subgroup

1. Purpose

The purpose of this White Paper is to demonstrate practices related to setting up and maintaining large project/program integrated master schedules (IMS) that include subprojects that are required to be individually submitted into the DOE Project Assessment and Reporting System (PARS). Even though there are subprojects in the IMS that do not report into PARS, it is recommended that those projects consider the outlined practices as they affect the reporting criteria of the PARS subprojects.

DOE is transitioning to a new format for data upload to PARS, using JavaScript Object Notation (JSON) dataset format. The new JSON format requirement, which will replace the existing MS Access file and CSV flat files, are documented within the DOE Contractor Project Performance (CPP) Data Item Description (DID). The new JSON dataset will be required for future PARS data uploads. The latest version of the DOE CPP DID is found at the PARS DOE website link.

PARS emphasis is to improve/streamline management of the scope and integration of WBS, schedule, and costs. The focus in the outlined practices is to illustrate the necessary steps required when certain DOE offices/sites maintain an IMS (enterprise site/portfolio/program) that comprise multiple inter-related PARS projects. The individual capital asset projects, each having unique PARS IDs are required to be reported separately in PARS, with each project able to stand alone in the tenets of schedule health and critical path analysis, or identification of tasks with total float less than or equal to zero, within PARS when segregated from the Program IMS. This requires that the submitted schedules must match the same information in the IMS for critical path, date and float calculations, resources, and interproject dependencies (driving and non-driving) originating from external projects included in the site/portfolio/program IMS. To accomplish this, the separated schedules must include interface milestones representing predecessor activities of external projects that link to the activities in the submitted PARS project.

The separation of the schedules for accurate reporting of stand-alone schedule files is facilitated by implementation of a process and methodology for development and maintenance of interface milestones. The standard traditional practice requires manual creation of interface milestones within the submitted PARS project. The use of interface milestones, as discussed later in this document, is necessary to facilitate the ability to create native stand-alone P6 schedule backups (e.g., XER/XML) files for importing into external P6 systems for review and analysis.

In addition to the necessary standard practice of developing and maintaining interface milestones, DOE has commissioned Oracle to develop automated capabilities in P6 to support the DOE CPP JSON DID for reporting project data into PARS.

The first Oracle P6 capability, herein referenced as “project-from-program”, automates generation of external interface milestones/tasks as a feature during the DOE CPP JSON export. With this capability, interface milestones are automatically created in the output of the JSON export schedule files. Although, this automated feature meets the requirement of PARS reporting of external dependencies in the JSON datasets, it is still a necessary practice for projects to consider manually creating interface milestones, as described later in this white paper.

The second Oracle P6 capability, part of the project-from-program feature, supports sites to maintain their unique PARS ID project schedule in separate multiple interlinked individual P6 schedule files. In this case, the new CPP JSON export feature will combine the reporting of the individual P6 schedules into one JSON dataset for that unique PARS ID project.

2. Setup Process

There are various setup methods that sites may implement as a process to managing a project/program integrated master portfolio schedule. The most common method is establishing an IMS for interrelated Capital Asset Projects, all being reported separately in PARS, with each project having its own unique PARS ID. Other methods may include an integrated master portfolio schedule that is a mix of Capital Asset Projects, Major Items of Equipment (MIE), General Plant Projects (GPP), Science, Environmental, other site-specific work, or even external DOE projects. It may include a mixture of federal, prime contractors, and subcontractor schedules. Regardless of the setup method, and for the subject of this white paper, an IMS will contain Capital Asset Projects that have external relationships and require individual submittal in PARS.

Rigorous application of the scheduling process in managing the interface relationships between projects is necessary to ensure that each project schedule (unique PARs project ID) can, and must, stand alone when separated from the master project portfolio schedule.

Oracle P6 uses the term “activity” while other documentations use the term “task”. In this white paper, both terms are used interchangeably.

The term internal (current) project schedule is used throughout this white paper to represent the reported project schedule in PARS (i.e. a project that has a unique PARS ID). The external project schedule represents a project schedule in the portfolio IMS that has scope outside the internal (current) project but has impact (logical dependencies) to the internal (current) project.

2.1. Schedule Structure / WBS Hierarchy

2.1.1. Schedule Hierarchy and Schedule Organization

In a multi-project portfolio (site or program), schedules for each project within the portfolio are developed to include only the scope of work associated with that project, aligning scope, schedule, and budget. External scope (outside the scope of that project) is modeled using schedule visibility tasks (i.e., milestones or tasks with duration) to represent the duration elapsed for the external scope or to fix a point in time (using constraint dates) for the start or finish of the external work that typically precedes tasks in the current project scope. Typically, external scope that is represented by these SVTs is directly correlated scope to the project but performed by entities outside the scope of that project (i.e., DOE approvals, government furnished equipment, etc.). Normally, SVTs are updated and maintained manually as dates and duration information of the external scope becomes available.

In addition to the planned SVT activities within each project (unique PARS ID), incoming (predecessor) external dependencies from projects contained in the integrated master portfolio schedule can be added as interface milestones (i.e. predecessor hand off milestones), as explained later in this white paper.

2.1.2. Integrated Master Program Level – Level 1

In a multi-portfolio or program environment, the structuring of the P6 Enterprise Project Structure (EPS) database is critical to facilitate tracking the entire portfolio/program (master) schedule. The master project schedule level is the EPS node of the P6 database. It is not necessary to develop an independent P6 master summary-level schedule, as all the schedule summarizations to level 1 are facilitated by coding (EPS structure coding, project coding, activity coding, etc.). It is best practice to account for schedule rollup to level 1 of the EPS hierarchy when setting up the P6 master schedule portfolio and P6 EPS nodes.

2.1.3. Subproject Schedule (Unique PARS ID) – Level 2 and Below

Each project with a unique PARS ID is considered a standalone schedule with complete scope for this project which is usually developed as one P6 schedule file. For practical purposes, this PARS ID project schedule would be considered level 2 in a multi-portfolio or program environment, with level 1 being the overall portfolio or program schedule represented as the highest level of the P6 EPS structure. The level 1 master integrated schedule is accessed when all project schedules are opened at the same time in the EPS highest level node. For tracking the portfolio/program level master schedule, the level 1 schedule can then be summarized into higher level activities, if needed, through the EPS structure, WBS or special activity coding. This type of summarization works well as it represents the actual detailed subproject information contained in each level 2 (subprojects) schedules, avoiding developing unnecessary high-level summary activities (i.e. hammocks).

2.1.4. Subprojects Inter-Project relationships

In a multi-project portfolio where there are several individual P6 files that interface with each other, inter-project relationships become a solution to dynamically plan, schedule, and demonstrate timely impacts to driving and critical paths.

Managing interproject relationships requires rigorous coordination amongst the individual project teams to ensure synchronicity between the individual project schedules such that impacts to current schedule data is accurately represented between projects.

2.1.5. Interface Points (Interface Milestones)

The recommended practice is to use interface milestones (i.e. hand off milestones) that reside in the internal (current) project schedule that link incoming external predecessor relationships to the interface milestones and then link the interface milestones to the task(s) in the current project schedule. Interface milestones, as a pass through for linking activities between external schedules activities and internal (current) schedule activities, is best practice to organize external relationships and manage inter-project links. This ensures visible and clear logic interface where the interface milestones will always represent intermediary points of intersection where predecessor(s) are task(s) for external scope and successor(s) are task(s) for internal (current) project scope. The interface milestones ensure rigor around configuration control during monthly status updates and forecasting. During the schedule update cycle, these interface milestones will be free flowing and dynamic. At month-end, after the integrated master portfolio schedule is scheduled (calculated), soft constraints will be applied to the interface milestones to hold their dates in time, which will represent the driving (or non-driving) external predecessor dates. This is important as once each project schedule is separated from the master schedule (the portfolio), the interface milestones (constrained) will hold the integrity of the project schedule and represent driving logic, essential for schedule calculations in the internal (current) project schedule.

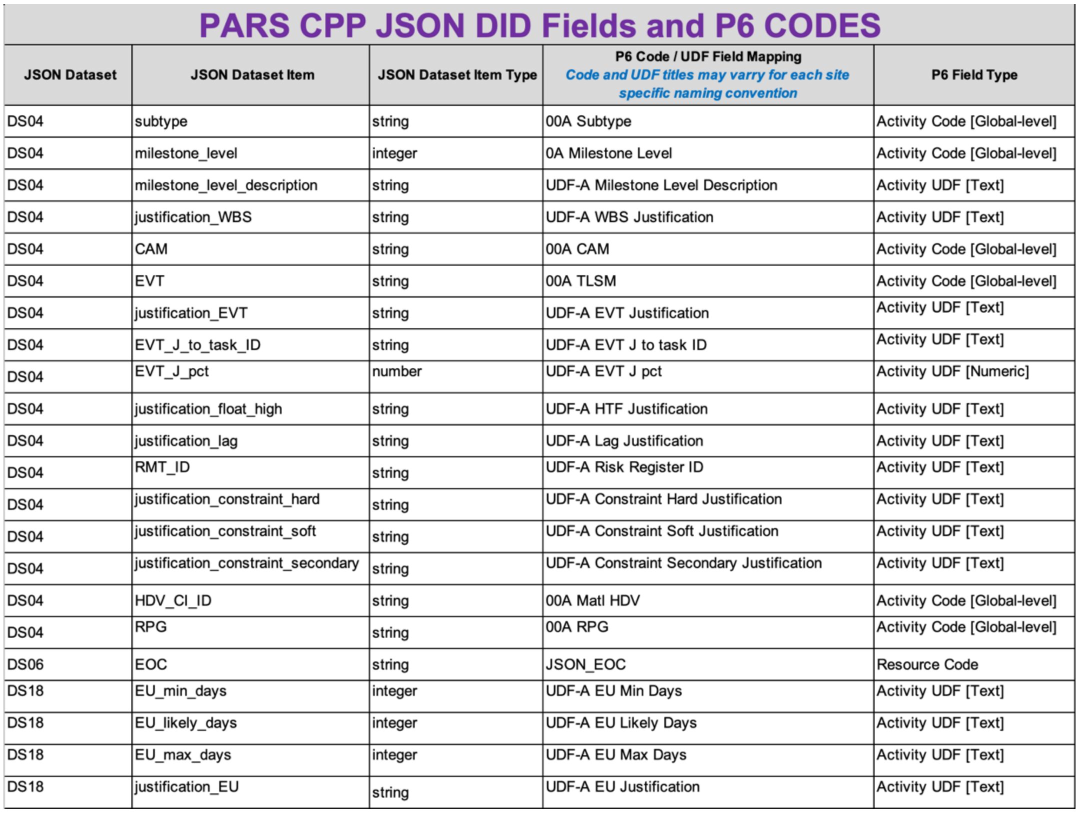
It is also common to have outgoing interface milestones from the internal (current schedule) to external schedules. In this case, these milestones are used for information and organization, to indicate that certain internal (current) schedule activities linked through the outgoing interface milestones are needed as predecessors to other external scope activities, contained outside the internal (current) project schedule. For the intended purpose of this white paper, only incoming interface milestones are the subject of discussion.

2.2. Schedule Coding

2.2.1. Standard Coding Requirements for Master/Subprojects Integration

It is recommended to review and align with the DOE CPP DID before planning the coding structure for the integrated master portfolio schedule. The new JSON dataset code structure should be propagated across the entire portfolio/program (master) schedule and its subprojects to ensure accurate reporting at the control account and work package levels. Special attention should be made to the CPP DID item “milestone_level”, as it identifies key tasks and milestones to be assigned a predefined code value per DID.

Each of the new JSON datasets has been identified in the JSON CPP DID with their specific field name, description, data type, and is identified if it is a required or key field. Most of the schedule data items (DS04, DS05, DS06, DS18, DS19, DS20) are locked P6 native standard fields (i.e. Early Start and Finish dates, etc.) that do not require field mapping. Other fields are custom fields (i.e. EVT, CAMs, Justifications, etc.) that require mapping. The following table in figure 1 lists the JSON custom fields, as defined in the CPP DID, that require mapping to custom P6 Code Fields or User Defined Fields to consider during schedule setup and implementation. The P6 mapped fields in this table is an example, each site will vary and set up their custom fields according to their master coding dictionary.



PARS CPP JSON DID Fields and P6 CODES

JSON Dataset	JSON Dataset Item	JSON Dataset Item Type	P6 Code / UDF Field Mapping *Code and UDF titles may varry for each site specific naming convention*	P6 Field Type
DS04	subtype	string	00A Subtype	Activity Code [Global-level]
DS04	milestone_level	integer	0A Milestone Level	Activity Code [Global-level]
DS04	milestone_level_description	string	UDF-A Milestone Level Description	Activity UDF [Text]
DS04	justification_WBS	string	UDF-A WBS Justification	Activity UDF [Text]
DS04	CAM	string	00A CAM	Activity Code [Global-level]
DS04	EVT	string	00A TLSM	Activity Code [Global-level]
DS04	justification_EVT	string	UDF-A EVT Justification	Activity UDF [Text]
DS04	EVT_J_to_task_ID	string	UDF-A EVT J to task ID	Activity UDF [Text]
DS04	EVT_J_pct	number	UDF-A EVT J pct	Activity UDF [Numeric]
DS04	justification_float_high	string	UDF-A HTF Justification	Activity UDF [Text]
DS04	justification_lag	string	UDF-A Lag Justification	Activity UDF [Text]
DS04	RMT_ID	string	UDF-A Risk Register ID	Activity UDF [Text]
DS04	justification_constraint_hard	string	UDF-A Constraint Hard Justification	Activity UDF [Text]
DS04	justification_constraint_soft	string	UDF-A Constraint Soft Justification	Activity UDF [Text]
DS04	justification_constraint_secondary	string	UDF-A Constraint Secondary Justification	Activity UDF [Text]
DS04	HDV_CI_ID	string	00A Matl HDV	Activity Code [Global-level]
DS04	RPG	string	00A RPG	Activity Code [Global-level]
DS06	EOC	string	JSON_EOC	Resource Code
DS18	EU_min_days	integer	UDF-A EU Min Days	Activity UDF [Text]
DS18	EU_likely_days	integer	UDF-A EU Likely Days	Activity UDF [Text]
DS18	EU_max_days	integer	UDF-A EU Max Days	Activity UDF [Text]
DS18	justification_EU	string	UDF-A EU Justification	Activity UDF [Text]

Figure 1: PARS CPP JSON Fields and Mapped P6 Codes / UDFs

2.2.2. Coding Requirements for Interface Milestones

Sites will establish their specific master coding dictionary that has specific activity/EPS code requirements for managing the various site project/program schedules as well as integration between systems. Interface milestones are considered SVTs and should be coded appropriately. It is recommended that interface milestones have special code requirements to identify them and facilitate grouping and filtering during schedule status updates and PARS reporting. This will also allow clear identification of the master and subproject(s) critical/driving paths and schedule analysis.

2.2.3. Scheduling of Subprojects

2.2.3.1. Common Data Date

It is recommended that a common data date for all master and subprojects be observed. This ensures schedule integrity and will minimize any variances observed due to a difference in the data date.

2.2.3.2. Frequency of Updates

Updates to the schedule are determined by the site/program system description and project/program requirements. Internal updates to the schedule can happen as frequently as needed for analysis and planning but is typically weekly or bi-weekly. For official schedule performance reporting, schedules are updated at the end of each monthly financial reporting period.

2.2.4. Monthly Business Rhythm Cycle

Successful projects and programs establish, execute, and follow a monthly business rhythm for the status, maintenance, and analysis of the IMS. The team creates this rhythm by establishing and following a standard business rhythm calendar that lays out regular schedule updates and project/program status reviews. Month-end (data date) is the pivotal point in time that sets the deliverables due dates in the business rhythm calendar. A key purpose for the business rhythm calendar is to ensure that intermediate products are available for review and validation. This ensures the integrity of all the interrelated systems and dependencies (i.e. interface milestones) are present. The business rhythm calendars broaden the project schedule visibility, providing regularly scheduled internal deadlines associated with managing the overall project/program schedule status updates and reporting (internal and external). It is important to include products and deadlines in the business rhythm calendar associated with managing interface milestones in a multi-project portfolio (site or program) environment. Establishing this rhythm early in the program life cycle provides the discipline and sense of importance necessary to motivate the project/program teams to collaboratively work to the project schedule and complete work regularly and on time. Each project with a unique PARS ID shall have a monthly business rhythm calendar identified and documented. It is a team effort to maintain the business rhythm calendar (Program Executive, Project Manager (PM), Control Account Manager (CAM), Planning and Finance). All disciplines must work together to ensure key deadlines are met.

The month-end process when managing in a multi-project portfolio (site or program) environment should include, but is not limited to:

- Status deadline – Identifies the date the status is completed and available for integration with Interface Milestones, or SVTs.
- Change control deadline – Identifies the date that changes can no longer be made to the project, or subproject.
- Critical Path Analysis – Identifies the date the critical path is locked down and identified in the project schedule (project and subproject, if required)
- Month-end validation
    - Baseline and Forecast start/finish dates.
    - Interface Milestones identified and coded.
    - Constraint date added.
    - Constraint justification identified and populated.

  - Critical path (project and subproject, if required) is reviewed and validated with PM(s) and project teams.
- Date the Interface Milestones are to be integrated and the JSON file to be prepared (discussed later in this paper.)

2.2.5. Baseline Change Control

Change control preserves the integrity of the performance measurement baseline (PMB) by formally controlling and properly documenting changes using a systematic approach. It ensures the PMB reflects the most current plan for accomplishing the project, thus providing credible performance measurement data on which the contractor and government can rely to make decisions. Schedule changes follow a formal baseline change control process that requires transparency regarding changes. Documentation is required to reflect the schedule condition before the requested change and after the change, and the rationale provides management sufficient visibility when reviewing and approving the change. It is important to ensure that the baseline is maintained through change control consistent with program requirements and the applicable system description.

The monthly change control process in a multi-project portfolio (site or program) environment that includes “sub-projects” needs to be defined and documented to ensure interface milestones between subprojects align and the data is consistent. When a change is identified at the sub-project level, the monthly business rhythm cycle for the sub-project identifies the change control deadline. Each subproject manages the change control consistent with the business rhythm of the subproject. When interface milestones dates are changed, the subproject team is responsible for making the updates before the month-end data is finalized, consistent with the change control process. Baseline changes to the interface milestones should be documented consistent with the month-end process for change at the master project and sub-project level. Following an established configuration management and a formal change control process when maintaining the baseline project schedule is paramount.

3. Maintenance

3.1. Schedule Update Process

On a recurring monthly cycle, the schedule is statused for internal (project) and external (DOE) reporting. Typically, each subproject (unique PARS ID) has an Integrated Project Team (IPT) (PM, CAMs, Project Controls staff, etc.) specific to that project that is responsible for status updates and reporting their project’s schedule information. As interface milestones are dynamically updated from external subprojects, it is important to coordinate with the external subprojects to ensure their updates are finalized, so the interface milestone dates are locked and held for the month-end reporting. Utilizing user-defined field (UDF) or activity code field to code the interface milestones will facilitate easier schedule updates and determination of schedule impacts from external dependencies.

The forecast schedule is calculated monthly using a status date (i.e. data date), typically established based on the site month-end financial calendar. The baseline schedule is also calculated monthly, if approved changes are incorporated.

3.2. Update of Interface Milestones (Dates / Logic)

During the monthly update cycle, the soft constraints are removed from the interface milestones to allow the subproject schedules in the master portfolio to move freely with the schedule logic as the schedules are calculated. This process can be automated (e.g. globally changed) with the use of P6 global change feature. New interface milestones can be added in the schedule, but attention should be made to the monthly business rhythm cycle to allow for the same interface milestones to be added in the baseline schedules, under change control.

As each subproject completes their updates, the interface milestones become static, and their dates are held for the month-end reporting. At this point, as external dependencies and their impacts are established, a soft constraint (Start on or After or Finish on or After) is applied to the interface milestones to pin the dates and finalize the month-end status update and forecast. The P6 global change feature can automate applying the soft constraints.

3.3. Evaluating Impact Due to External Dependencies

In this document, the multi-project portfolio represents linked subprojects though inter-project relationships, connected directly between the subproject activities or using interface milestones (preferred method). Therefore, it is expected as schedules are updated monthly, elaborate analysis of impact should be performed due to the dynamic master/subproject links and progress of activities.

During the update, impacts of external dependencies, through the interface milestones, are identified. The interface milestones are the key points (intermediary) of interface between external and internal activities, therefore, the dates of those interface milestones can be compared to the previous month to help isolate and analyze impacts for external driving logic and their impact on internal (current) subproject. Advanced export/import features within P6 coupled with established monthly update process, can expedite, and facilitate efficient management of the interface milestone updates and their driving logic impacts.

Communication with enterprise IPT and subproject IPTs should be consistently maintained to fully understand the impacts of external subproject to the internal (current) project. Performing monthly critical path and near critical path analyses of these dependencies will help raise early warnings to the internal IPT and should be discussed monthly to prevent or prepare for risks and issues as they arise.

4. Reporting

Under DOE O 413.3B, contractors are required to submit performance data electronically to DOE via PARS, found at https://json.pars.doe.gov. Contractors must submit monthly project performance data no later than CD-2 for each unique PARS ID project having a total project cost (TPC) greater than $50M. Each unique PARS ID project must be submitted separately as a standalone project. Existing format is documented in published DOE CPP Upload Requirements for PARS. The main PARS submittal includes cost and schedule datasets submitted in CSV flat files, other format using MS Access file is also available on limited basis for certain sites. DOE will be replacing MS Access file and CSV flat file formats with JSON format in calendar year 2025. DOE CPP JSON upload requirements are documented in DID which outlines the various datasets (files) and their data elements.

The current schedule CSV flat file format includes schedule activity information, logic relationships, and resource assignment based on the P6 schedule for the unique project. Similarly, the new JSON format will include the same information, but mainly can be generated directly from the software tools (P6 for schedule data and Cobra, WebEVM, and forProject for cost data, as examples).

4.1. Export (XER/XML) Subprojects Individually to PARS (unique PARS ID)

Schedule files in native P6 format are submitted to DOE HQ via the DOE field office. Each project with a unique PARS ID is considered a standalone project, therefore, the P6 schedule file submitted for that project, must also represent a standalone schedule with its own critical path. This means, when the project schedule is separated from the master portfolio and exported in XER or XML format, it must be unchanged when restored by DOE as an individual standalone schedule (i.e. same critical path, dates, float calculations, etc.). As explained earlier in this document, applying the soft constraint dates for the interface milestones ensures that external dependencies are accounted for and their dates are held, even after the schedule is separated from the master portfolio and restored in a separate P6 environment (DOE).

4.2. Export Individual Subproject Schedules in JSON Dataset Files

The DOE CPP DID requires that each project schedule with a unique PARS ID is submitted in a specific format. As part of this format, the external dependencies are required to be shown as SVTs from the various external project schedules. These SVTs will have the P6 schedule file name to which they belong and will be contained in the JSON data item entitled “subproject_ID. Oracle has automated this feature as part of the DOE CPP JSON export built within P6, as described in the section below.

5. Oracle Automated Interface Milestone Feature in P6

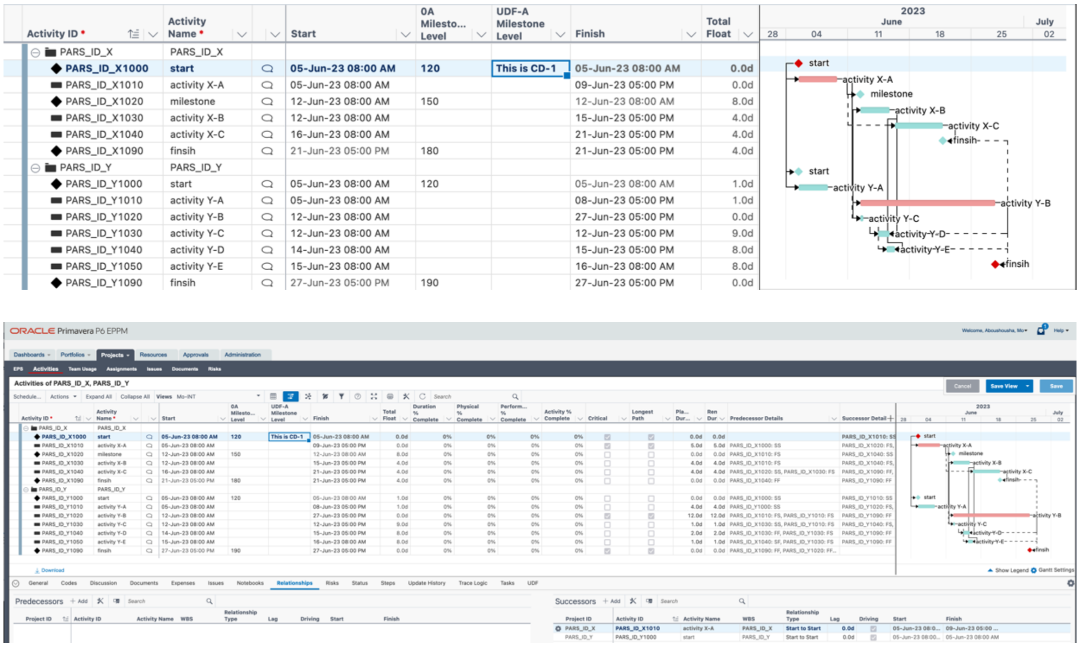
Oracle had originally added a feature in P6 releases (P6 Web version 23.12 and P6 EPPM On Premise version 22.12.2 and later) to automate generation of external interface milestone/tasks as part of the DOE CPP JSON dataset files. Oracle later added this feature in P6 Professional version 23.12. Oracle has implemented software enhancements in support of this feature since the original release, with current features added in subsequent P6 releases.

This added feature allows for the inclusion of automatically generated SVTs representing the external tasks and their logical relationships to the tasks in the schedule being exported when projects use the export template embedded in P6 for exporting the JSON datasets. These SVT external interface tasks will be created only in the JSON schedule dataset (DS04) file (not in the actual P6 file) and will be tagged with metadata that includes unique activity ID, activity descriptions designating the SVT marking, start/finish dates, constraints, appropriate coding, WBS, P6 subproject ID, and other relevant parameters. The external SVTs will have designated flags for driving / non-driving dependencies from the external project predecessor(s) tasks to the PARS project being submitted. All logical relationships from external SVTs to the internal (current) PARS project will also be identified in the JSON logic dataset (DS05) file.

To demonstrate the Oracle P6 capability, consider the following example:

Assume two projects that comprise the site/portfolio/program IMS, project “PARS_ID_X” and project “PARS_ID_Y”, refer to figure 2. These two projects are individual P6 files with unique PARS ID and reside in the master project level EPS node. For simplicity the first project will be referenced as Proj-X and the second as Proj-Y.

There are five activities/milestones in Proj-X that have inter-project relationships to six activities/milestones in Proj-Y. There is one activity in Proj-X (PARS_ID_X1030) that has relationships to two activities in Proj-Y (PARS_ID_Y1030 and PARS_ID_Y1040). Also, as shown in figure 2, one of the activities in Proj-X (PARS_ID_X1010) is on the driving and critical path, driving activity PARS_ID_Y1020 in Proj-Y.



Activity ID	Activity Name	Start	0A Milesto... Level	UDF-A Milestone Level	Finish	Total Float
PARS_ID_X	PARS_ID_X					
PARS_ID_X1000	start	05-Jun-23 08:00 AM	120	This is CD-1	05-Jun-23 08:00 AM	0.0d
PARS_ID_X1010	activity X-A	05-Jun-23 08:00 AM			09-Jun-23 05:00 PM	0.0d
PARS_ID_X1020	milestone	12-Jun-23 08:00 AM	150		12-Jun-23 08:00 AM	8.0d
PARS_ID_X1030	activity X-B	12-Jun-23 08:00 AM			15-Jun-23 05:00 PM	4.0d
PARS_ID_X1040	activity X-C	16-Jun-23 08:00 AM			21-Jun-23 05:00 PM	4.0d
PARS_ID_X1090	finsih	21-Jun-23 05:00 PM	180		21-Jun-23 05:00 PM	4.0d
PARS_ID_Y	PARS_ID_Y					
PARS_ID_Y1000	start	05-Jun-23 08:00 AM	120		05-Jun-23 08:00 AM	1.0d
PARS_ID_Y1010	activity Y-A	05-Jun-23 08:00 AM			08-Jun-23 05:00 PM	1.0d
PARS_ID_Y1020	activity Y-B	12-Jun-23 08:00 AM			27-Jun-23 05:00 PM	0.0d
PARS_ID_Y1030	activity Y-C	12-Jun-23 08:00 AM			12-Jun-23 05:00 PM	9.0d
PARS_ID_Y1040	activity Y-D	14-Jun-23 08:00 AM			15-Jun-23 05:00 PM	8.0d
PARS_ID_Y1050	activity Y-E	15-Jun-23 08:00 AM			16-Jun-23 08:00 AM	8.0d
PARS_ID_Y1090	finsih	27-Jun-23 05:00 PM	190		27-Jun-23 05:00 PM	0.0d

Activities of PARS_ID_X, PARS_ID_Y

Activity ID	Activity Name	Start	0A Milesto... Level	UDF-A Milestone Level	Finish	Total Float	Duration % Complete	Physical % Complete	Perform... % Complete	Activity % Complete	Critical	Longest Path	Pla... Dur...	Ren Dur	Predecessor Details	Successor Detail
PARS_ID_X	PARS_ID_X															
PARS_ID_X1000	start	05-Jun-23 08:00 AM	120	This is CD-1	05-Jun-23 08:00 AM	0.0d	0%	0%	0%	0%	☑	☑	0.0d	0.0d		PARS_ID_X1010: SS
PARS_ID_X1010	activity X-A	05-Jun-23 08:00 AM			09-Jun-23 05:00 PM	0.0d	0%	0%	0%	0%	☑	☑	5.0d	5.0d	PARS_ID_X1000: SS	PARS_ID_X1020: FS,
PARS_ID_X1020	milestone	12-Jun-23 08:00 AM	150		12-Jun-23 08:00 AM	8.0d	0%	0%	0%	0%	□	□	0.0d	0.0d	PARS_ID_X1010: FS	PARS_ID_X1040: SS
PARS_ID_X1030	activity X-B	12-Jun-23 08:00 AM			15-Jun-23 05:00 PM	4.0d	0%	0%	0%	0%	□	□	4.0d	4.0d	PARS_ID_X1010: FS	PARS_ID_X1040: FS,
PARS_ID_X1040	activity X-C	16-Jun-23 08:00 AM			21-Jun-23 05:00 PM	4.0d	0%	0%	0%	0%	□	□	4.0d	4.0d	PARS_ID_X1020: SS, PARS_ID_X1030: FS	PARS_ID_X1090: FF,
PARS_ID_X1090	finsih	21-Jun-23 05:00 PM	180		21-Jun-23 05:00 PM	4.0d	0%	0%	0%	0%	□	□	0.0d	0.0d	PARS_ID_X1040: FF	PARS_ID_Y1090: FF
PARS_ID_Y	PARS_ID_Y															
PARS_ID_Y1000	start	05-Jun-23 08:00 AM	120		05-Jun-23 08:00 AM	1.0d	0%	0%	0%	0%	□	□	0.0d	0.0d	PARS_ID_X1000: SS	PARS_ID_Y1010: SS
PARS_ID_Y1010	activity Y-A	05-Jun-23 08:00 AM			08-Jun-23 05:00 PM	1.0d	0%	0%	0%	0%	□	□	4.0d	4.0d	PARS_ID_Y1000: SS	PARS_ID_Y1020: FS,
PARS_ID_Y1020	activity Y-B	12-Jun-23 08:00 AM			27-Jun-23 05:00 PM	0.0d	0%	0%	0%	0%	☑	☑	12.0d	12.0d	PARS_ID_X1010: FS, PARS_ID_Y1010: FS	PARS_ID_Y1090: FF
PARS_ID_Y1030	activity Y-C	12-Jun-23 08:00 AM			12-Jun-23 05:00 PM	9.0d	0%	0%	0%	0%	□	□	1.0d	1.0d	PARS_ID_X1030: SS, PARS_ID_Y1010: FS	PARS_ID_Y1040: FS,
PARS_ID_Y1040	activity Y-D	14-Jun-23 08:00 AM			15-Jun-23 05:00 PM	8.0d	0%	0%	0%	0%	□	□	2.0d	2.0d	PARS_ID_X1030: FF, PARS_ID_Y1030: FS	PARS_ID_Y1090: FF
PARS_ID_Y1050	activity Y-E	15-Jun-23 08:00 AM			16-Jun-23 08:00 AM	8.0d	0%	0%	0%	0%	□	□	1.0d	1.0d	PARS_ID_X1040: SF, PARS_ID_Y1030: FS	PARS_ID_Y1090: FF
PARS_ID_Y1090	finsih	27-Jun-23 05:00 PM	190		27-Jun-23 05:00 PM	0.0d	0%	0%	0%	0%	☑	☑	0.0d	0.0d	PARS_ID_X1090: FF, PARS_ID_Y1020: FF...	

Predecessors

Project ID	Activity ID	Activity Name	WBS	Relationship Type	Lag	Driving	Start	Finish

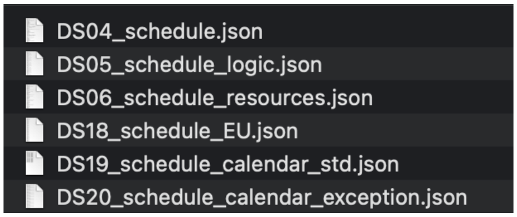
Successors

Project ID	Activity ID	Activity Name	WBS	Relationship Type	Lag	Driving	Start	Finish
PARS_ID_X	PARS_ID_X1010	activity X-A	PARS_ID_X	Start to Start	0.0d	☑	05-Jun-23 08:0...	09-Jun-23 05:00 ...
PARS_ID_Y	PARS_ID_Y1000	start	PARS_ID_Y	Start to Start	0.0d	☑	05-Jun-23 08:00...	05-Jun-23 08:00 AM

Figure 2: P6 layout with inter-project relationships between Proj-X and Proj-Y

Assume Proj-Y is the project that is to be exported for reporting to PARS. The DOE CPP JSON export is run from P6, the export will generate 6 JSON DS files, as shown in figure 3 and according to the DOE CPP DID.

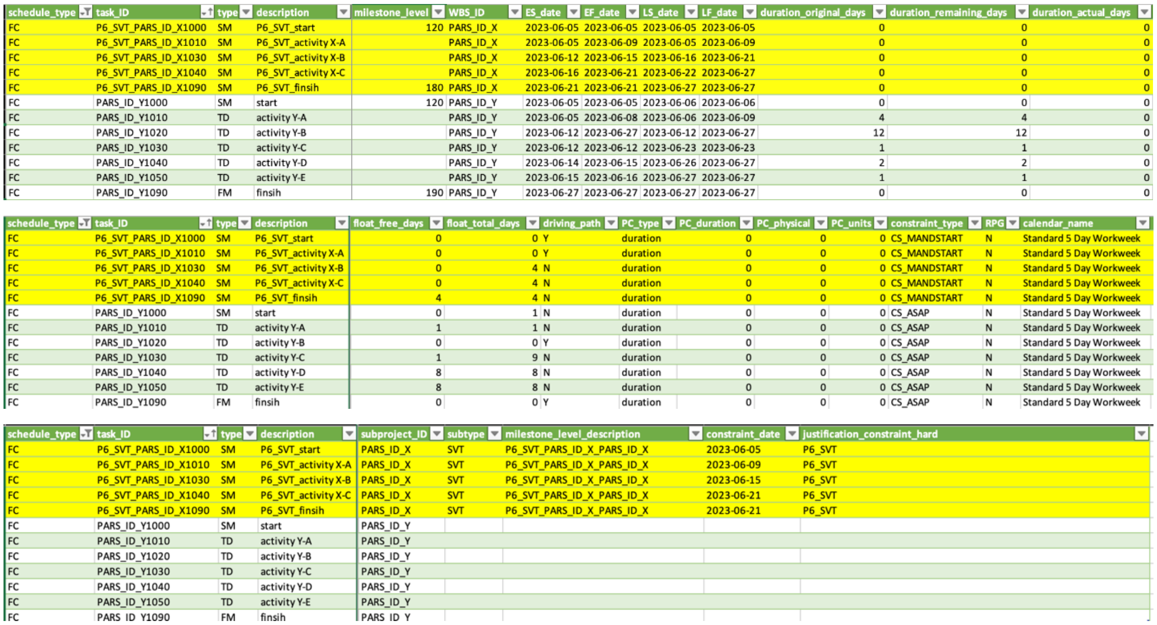
The files will include various schedule information for Proj-Y that include data for activities, logic, resources, estimate uncertainty, and activity calendars.



DS04_schedule.json
DS05_schedule_logic.json
DS06_schedule_resources.json
DS18_schedule_EU.json
DS19_schedule_calendar_std.json
DS20_schedule_calendar_exception.json

Figure 3: JSON Schedule DS files generated from P6

The JSON exports will automatically generate the five external activities of Proj-X in the JSON schedule output file (DS04) for Proj-Y. These activities will be shown as SVTs with start constraint dates to pin them in place, and indicating their total and free float values, and driving path designation to the PARS project (Proj-Y). The SVTs will have unique activity ID and description and will be coded as “SVT” in the subtype data item code field. The SVTs will have other data items and code fields automatically populated that include P6 subproject file ID where they belong, start/finish dates, calendar, constraint type and date, justification notes, WBS, milestone level description, and other relevant parameters, refer to the five highlighted rows in figure 4.



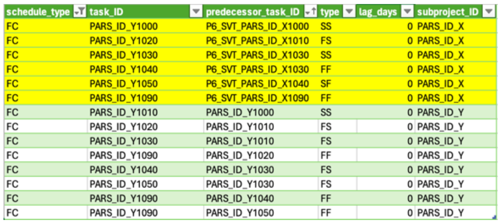
schedule_type	task_ID	type	description	milestone_level	WBS_ID	ES_date	EF_date	LS_date	LF_date	duration_original_days	duration_remaining_days	duration_actual_days
FC	P6_SVT_PARS_ID_X1000	SM	P6_SVT_start	120	PARS_ID_X	2023-06-05	2023-06-05	2023-06-05	2023-06-05	0	0	0
FC	P6_SVT_PARS_ID_X1010	SM	P6_SVT_activity X-A		PARS_ID_X	2023-06-05	2023-06-09	2023-06-05	2023-06-09	0	0	0
FC	P6_SVT_PARS_ID_X1030	SM	P6_SVT_activity X-B		PARS_ID_X	2023-06-12	2023-06-15	2023-06-16	2023-06-21	0	0	0
FC	P6_SVT_PARS_ID_X1040	SM	P6_SVT_activity X-C		PARS_ID_X	2023-06-16	2023-06-21	2023-06-22	2023-06-27	0	0	0
FC	P6_SVT_PARS_ID_X1090	SM	P6_SVT_finsih	180	PARS_ID_X	2023-06-21	2023-06-21	2023-06-27	2023-06-27	0	0	0
FC	PARS_ID_Y1000	SM	start	120	PARS_ID_Y	2023-06-05	2023-06-05	2023-06-06	2023-06-06	0	0	0
FC	PARS_ID_Y1010	TD	activity Y-A		PARS_ID_Y	2023-06-05	2023-06-08	2023-06-06	2023-06-09	4	4	0
FC	PARS_ID_Y1020	TD	activity Y-B		PARS_ID_Y	2023-06-12	2023-06-27	2023-06-12	2023-06-27	12	12	0
FC	PARS_ID_Y1030	TD	activity Y-C		PARS_ID_Y	2023-06-12	2023-06-12	2023-06-23	2023-06-23	1	1	0
FC	PARS_ID_Y1040	TD	activity Y-D		PARS_ID_Y	2023-06-14	2023-06-15	2023-06-26	2023-06-27	2	2	0
FC	PARS_ID_Y1050	TD	activity Y-E		PARS_ID_Y	2023-06-15	2023-06-16	2023-06-27	2023-06-27	1	1	0
FC	PARS_ID_Y1090	FM	finsih	190	PARS_ID_Y	2023-06-27	2023-06-27	2023-06-27	2023-06-27	0	0	0

schedule_type	task_ID	type	description	float_free_days	float_total_days	driving_path	PC_type	PC_duration	PC_physical	PC_units	constraint_type	RPG	calendar_name
FC	P6_SVT_PARS_ID_X1000	SM	P6_SVT_start	0	0	Y	duration	0	0	0	CS_MANDSTART	N	Standard 5 Day Workweek
FC	P6_SVT_PARS_ID_X1010	SM	P6_SVT_activity X-A	0	0	Y	duration	0	0	0	CS_MANDSTART	N	Standard 5 Day Workweek
FC	P6_SVT_PARS_ID_X1030	SM	P6_SVT_activity X-B	0	4	N	duration	0	0	0	CS_MANDSTART	N	Standard 5 Day Workweek
FC	P6_SVT_PARS_ID_X1040	SM	P6_SVT_activity X-C	0	4	N	duration	0	0	0	CS_MANDSTART	N	Standard 5 Day Workweek
FC	P6_SVT_PARS_ID_X1090	SM	P6_SVT_finsih	4	4	N	duration	0	0	0	CS_MANDSTART	N	Standard 5 Day Workweek
FC	PARS_ID_Y1000	SM	start	0	1	N	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek
FC	PARS_ID_Y1010	TD	activity Y-A	1	1	N	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek
FC	PARS_ID_Y1020	TD	activity Y-B	0	0	Y	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek
FC	PARS_ID_Y1030	TD	activity Y-C	1	9	N	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek
FC	PARS_ID_Y1040	TD	activity Y-D	8	8	N	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek
FC	PARS_ID_Y1050	TD	activity Y-E	8	8	N	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek
FC	PARS_ID_Y1090	FM	finsih	0	0	Y	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek

schedule_type	task_ID	type	description	subproject_ID	subtype	milestone_level_description	constraint_date	justification_constraint_hard
FC	P6_SVT_PARS_ID_X1000	SM	P6_SVT_start	PARS_ID_X	SVT	P6_SVT_PARS_ID_X_PARS_ID_X	2023-06-05	P6_SVT
FC	P6_SVT_PARS_ID_X1010	SM	P6_SVT_activity X-A	PARS_ID_X	SVT	P6_SVT_PARS_ID_X_PARS_ID_X	2023-06-09	P6_SVT
FC	P6_SVT_PARS_ID_X1030	SM	P6_SVT_activity X-B	PARS_ID_X	SVT	P6_SVT_PARS_ID_X_PARS_ID_X	2023-06-15	P6_SVT
FC	P6_SVT_PARS_ID_X1040	SM	P6_SVT_activity X-C	PARS_ID_X	SVT	P6_SVT_PARS_ID_X_PARS_ID_X	2023-06-21	P6_SVT
FC	P6_SVT_PARS_ID_X1090	SM	P6_SVT_finsih	PARS_ID_X	SVT	P6_SVT_PARS_ID_X_PARS_ID_X	2023-06-21	P6_SVT
FC	PARS_ID_Y1000	SM	start	PARS_ID_Y				
FC	PARS_ID_Y1010	TD	activity Y-A	PARS_ID_Y				
FC	PARS_ID_Y1020	TD	activity Y-B	PARS_ID_Y				
FC	PARS_ID_Y1030	TD	activity Y-C	PARS_ID_Y				
FC	PARS_ID_Y1040	TD	activity Y-D	PARS_ID_Y				
FC	PARS_ID_Y1050	TD	activity Y-E	PARS_ID_Y				
FC	PARS_ID_Y1090	FM	finsih	PARS_ID_Y				

Figure 4: DS04_Schedule file for Proj-Y showing created SVTs from Proj-X

The exported JSON logic file (DS05) for Proj-Y will also show the six relationships (predecessors) of Proj-X, refer to highlighted rows in figure 5.



schedule_type	task_ID	predecessor_task_ID	type	lag_days	subproject_ID
FC	PARS_ID_Y1000	P6_SVT_PARS_ID_X1000	SS	0	PARS_ID_X
FC	PARS_ID_Y1020	P6_SVT_PARS_ID_X1010	FS	0	PARS_ID_X
FC	PARS_ID_Y1030	P6_SVT_PARS_ID_X1030	SS	0	PARS_ID_X
FC	PARS_ID_Y1040	P6_SVT_PARS_ID_X1030	FF	0	PARS_ID_X
FC	PARS_ID_Y1050	P6_SVT_PARS_ID_X1040	SF	0	PARS_ID_X
FC	PARS_ID_Y1090	P6_SVT_PARS_ID_X1090	FF	0	PARS_ID_X
FC	PARS_ID_Y1010	PARS_ID_Y1000	SS	0	PARS_ID_Y
FC	PARS_ID_Y1020	PARS_ID_Y1010	FS	0	PARS_ID_Y
FC	PARS_ID_Y1030	PARS_ID_Y1010	FS	0	PARS_ID_Y
FC	PARS_ID_Y1090	PARS_ID_Y1020	FF	0	PARS_ID_Y
FC	PARS_ID_Y1040	PARS_ID_Y1030	FS	0	PARS_ID_Y
FC	PARS_ID_Y1050	PARS_ID_Y1030	FS	0	PARS_ID_Y
FC	PARS_ID_Y1090	PARS_ID_Y1040	FF	0	PARS_ID_Y
FC	PARS_ID_Y1090	PARS_ID_Y1050	FF	0	PARS_ID_Y

Figure 5: DS05_Logic file for Proj-Y showing predecessor SVTs logic from Proj-X

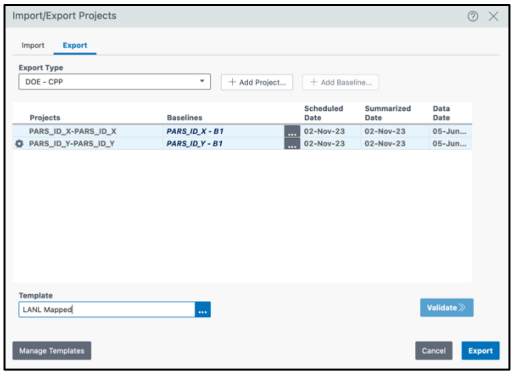
As can be interpreted in the above example, the new feature in Oracle P6 automates creation of the external SVT interface points and allows visibility of external dependencies to be represented in the internal (current) PARS project. Although, the Oracle P6 feature fulfills the requirement for reporting external dependencies (SVTs) in DOE CPP JSON schedule datasets for reporting to PARS, those SVTs (i.e. interface milestones) do not exist in the actual P6 schedule file. As explained earlier in this white paper, the interface milestones serve as an organized methodology to understand impact of external logical inter-dependencies and provide better management and control during schedule analysis and updates. Therefore, as a recommended standard practice, consideration should be made to include manually created interface milestones in the internal (current) PARS project.

6. Oracle Automated Feature in P6 To Combine Schedules in One JSON Datasets

There is another feature that Oracle implemented in the DOE CPP JSON export in P6 for combining separate P6 schedules in one JSON datasets for submitting to PARS. This feature is currently available in EPPM On Premise and P6 Professional 23.12 and later updates. The premise of this feature is to allow a site that is required to report its project as one unique PARS project schedule submittal, but its project schedule is separated into multiple (integrated) P6 schedule files. The new feature will allow the site to select all the P6 schedule files that comprise the PARS project schedule when generating the DOE CPP JSON export datasets. The JSON export will then generate one JSON schedule datasets (6 JSON dataset files) that encompasses schedule data for all the selected schedule files.

This is an important feature for a multi-site project where each site is developing and updating their part of the overall project schedule. In this example, the master schedule portfolio is the Level 1 EPS P6 node, which represents the unique PARS project, and the sites individual schedules are the subproject Level 2 schedule files, under the Level 1 EPS node. At month end, after all sites update their schedules, all P6 schedule files in the master portfolio (Level 1) are opened and the portfolio is scheduled to account for the integrated schedule, (including each site inter-project dependencies). For reporting to PARS, all the site schedule files are selected when running the DOE CPP JSON export to generate one JSON datasets for the one unique PARS project ID.

To demonstrate, consider the same example above, Proj-X and Proj-Y. Assume the project that comprises the one PARS ID is split into two integrated schedules (Proj-X and Proj-Y), that contain inter-project logic. When exporting the JSON datasets for the PARS ID project, the two projects are selected, see figure 6. The DOE CPP JSON export is then run within P6 (for the two projects together), the export will generate 6 JSON DS files, as demonstrated above in figure 3.



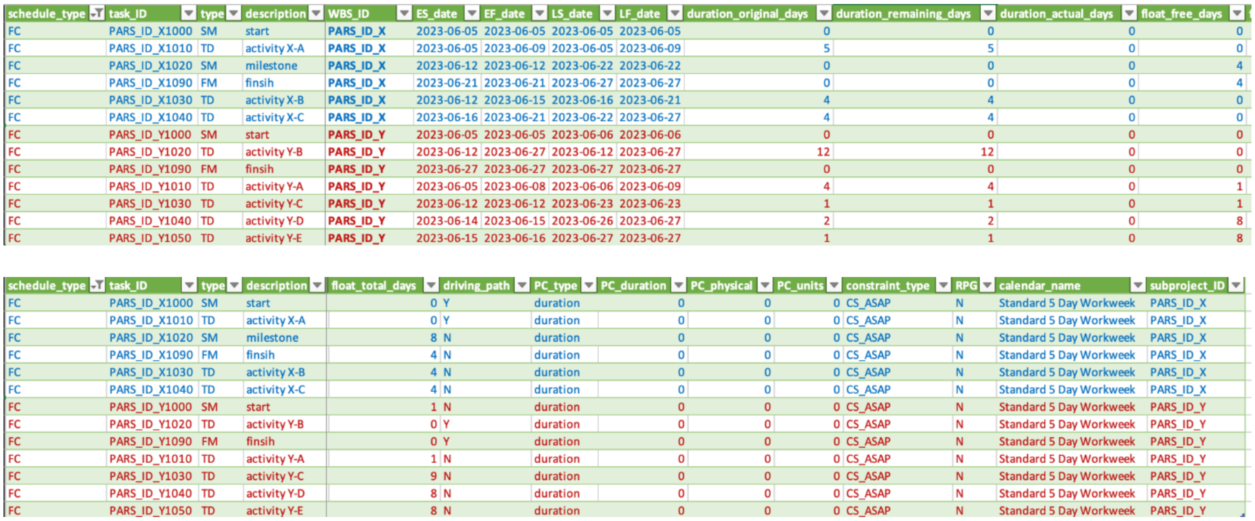
Export Type: DOE - CPP

Projects	Baselines	Scheduled Date	Summarized Date	Data Date
PARS_ID_X-PARS_ID_X	*PARS_ID_X - B1*	02-Nov-23	02-Nov-23	05-Jun...
PARS_ID_Y-PARS_ID_Y	*PARS_ID_Y - B1*	02-Nov-23	02-Nov-23	05-Jun...

Template: LANL Mapped

Figure 6: P6 layout with inter-project relationships between Proj-X and Proj-Y

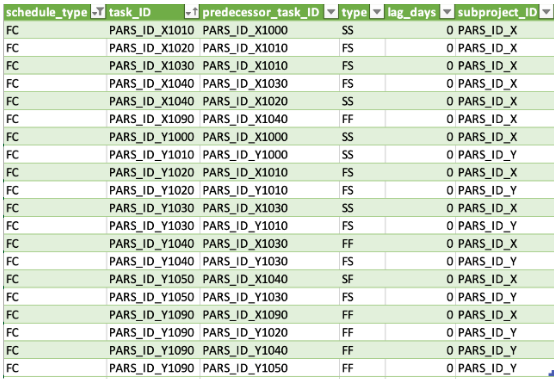
In this instance, as shown in figures 7 and 8, the generated dataset files will include combined various schedule information for both Proj-X and Proj-Y, including combined data for activities, logic, resources, estimate uncertainty, and activity calendars.



schedule_type	task_ID	type	description	WBS_ID	ES_date	EF_date	LS_date	LF_date	duration_original_days	duration_remaining_days	duration_actual_days	float_free_days
FC	PARS_ID_X1000	SM	start	**PARS_ID_X**	2023-06-05	2023-06-05	2023-06-05	2023-06-05	0	0	0	0
FC	PARS_ID_X1010	TD	activity X-A	**PARS_ID_X**	2023-06-05	2023-06-09	2023-06-05	2023-06-09	5	5	0	0
FC	PARS_ID_X1020	SM	milestone	**PARS_ID_X**	2023-06-12	2023-06-12	2023-06-22	2023-06-22	0	0	0	4
FC	PARS_ID_X1090	FM	finsih	**PARS_ID_X**	2023-06-21	2023-06-21	2023-06-27	2023-06-27	0	0	0	4
FC	PARS_ID_X1030	TD	activity X-B	**PARS_ID_X**	2023-06-12	2023-06-15	2023-06-16	2023-06-21	4	4	0	0
FC	PARS_ID_X1040	TD	activity X-C	**PARS_ID_X**	2023-06-16	2023-06-21	2023-06-22	2023-06-27	4	4	0	0
FC	PARS_ID_Y1000	SM	start	**PARS_ID_Y**	2023-06-05	2023-06-05	2023-06-06	2023-06-06	0	0	0	0
FC	PARS_ID_Y1020	TD	activity Y-B	**PARS_ID_Y**	2023-06-12	2023-06-27	2023-06-12	2023-06-27	12	12	0	0
FC	PARS_ID_Y1090	FM	finsih	**PARS_ID_Y**	2023-06-27	2023-06-27	2023-06-27	2023-06-27	0	0	0	0
FC	PARS_ID_Y1010	TD	activity Y-A	**PARS_ID_Y**	2023-06-05	2023-06-08	2023-06-06	2023-06-09	4	4	0	1
FC	PARS_ID_Y1030	TD	activity Y-C	**PARS_ID_Y**	2023-06-12	2023-06-12	2023-06-23	2023-06-23	1	1	0	1
FC	PARS_ID_Y1040	TD	activity Y-D	**PARS_ID_Y**	2023-06-14	2023-06-15	2023-06-26	2023-06-27	2	2	0	8
FC	PARS_ID_Y1050	TD	activity Y-E	**PARS_ID_Y**	2023-06-15	2023-06-16	2023-06-27	2023-06-27	1	1	0	8

schedule_type	task_ID	type	description	float_total_days	driving_path	PC_type	PC_duration	PC_physical	PC_units	constraint_type	RPG	calendar_name	subproject_ID
FC	PARS_ID_X1000	SM	start	0	Y	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek	PARS_ID_X
FC	PARS_ID_X1010	TD	activity X-A	0	Y	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek	PARS_ID_X
FC	PARS_ID_X1020	SM	milestone	8	N	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek	PARS_ID_X
FC	PARS_ID_X1090	FM	finsih	4	N	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek	PARS_ID_X
FC	PARS_ID_X1030	TD	activity X-B	4	N	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek	PARS_ID_X
FC	PARS_ID_X1040	TD	activity X-C	4	N	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek	PARS_ID_X
FC	PARS_ID_Y1000	SM	start	1	N	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek	PARS_ID_Y
FC	PARS_ID_Y1020	TD	activity Y-B	0	Y	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek	PARS_ID_Y
FC	PARS_ID_Y1090	FM	finsih	0	Y	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek	PARS_ID_Y
FC	PARS_ID_Y1010	TD	activity Y-A	1	N	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek	PARS_ID_Y
FC	PARS_ID_Y1030	TD	activity Y-C	9	N	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek	PARS_ID_Y
FC	PARS_ID_Y1040	TD	activity Y-D	8	N	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek	PARS_ID_Y
FC	PARS_ID_Y1050	TD	activity Y-E	8	N	duration	0	0	0	CS_ASAP	N	Standard 5 Day Workweek	PARS_ID_Y

Figure 7: DS04_Schedule file for combined Proj-X and Proj-Y



schedule_type	task_ID	predecessor_task_ID	type	lag_days	subproject_ID
FC	PARS_ID_X1010	PARS_ID_X1000	SS	0	PARS_ID_X
FC	PARS_ID_X1020	PARS_ID_X1010	FS	0	PARS_ID_X
FC	PARS_ID_X1030	PARS_ID_X1010	FS	0	PARS_ID_X
FC	PARS_ID_X1040	PARS_ID_X1030	FS	0	PARS_ID_X
FC	PARS_ID_X1040	PARS_ID_X1020	SS	0	PARS_ID_X
FC	PARS_ID_X1090	PARS_ID_X1040	FF	0	PARS_ID_X
FC	PARS_ID_Y1000	PARS_ID_X1000	SS	0	PARS_ID_X
FC	PARS_ID_Y1010	PARS_ID_Y1000	SS	0	PARS_ID_Y
FC	PARS_ID_Y1020	PARS_ID_X1010	FS	0	PARS_ID_X
FC	PARS_ID_Y1020	PARS_ID_Y1010	FS	0	PARS_ID_Y
FC	PARS_ID_Y1030	PARS_ID_X1030	SS	0	PARS_ID_X
FC	PARS_ID_Y1030	PARS_ID_Y1010	FS	0	PARS_ID_Y
FC	PARS_ID_Y1040	PARS_ID_X1030	FF	0	PARS_ID_X
FC	PARS_ID_Y1040	PARS_ID_Y1030	FS	0	PARS_ID_Y
FC	PARS_ID_Y1050	PARS_ID_X1040	SF	0	PARS_ID_X
FC	PARS_ID_Y1050	PARS_ID_Y1030	FS	0	PARS_ID_Y
FC	PARS_ID_Y1090	PARS_ID_X1090	FF	0	PARS_ID_X
FC	PARS_ID_Y1090	PARS_ID_Y1020	FF	0	PARS_ID_Y
FC	PARS_ID_Y1090	PARS_ID_Y1040	FF	0	PARS_ID_Y
FC	PARS_ID_Y1090	PARS_ID_Y1050	FF	0	PARS_ID_Y

Figure 8: DS05_Logic file for combined Proj-X and Proj-Y showing inter-project logic

As can be interpreted in the above example, the new feature in Oracle P6 will combine data from multiple schedules into one JSON CPP datasets for that one unique PARS project submittal.

7. Conclusion

The outlined practices in this white paper provide guidelines for management of an integrated master project/program schedules that comprise multi project schedules. The management of portfolio schedules requires rigor, coordination and discipline to provide accurate information to be reported to PARS. In addition, Oracle has introduced new capabilities in P6 that allows automated and seamless reporting of external inter-project dependencies in the new PARS CPP JSON datasets.