AzTech INTERNATIONAL
25132 Ocean Knoll Drive
Dana Point, California 92629
p: 949.249.9349
f: 949.363.9568
e: info@goAzTech.com

Run!AzTech Quick Look GASP Filters

Table of Contents

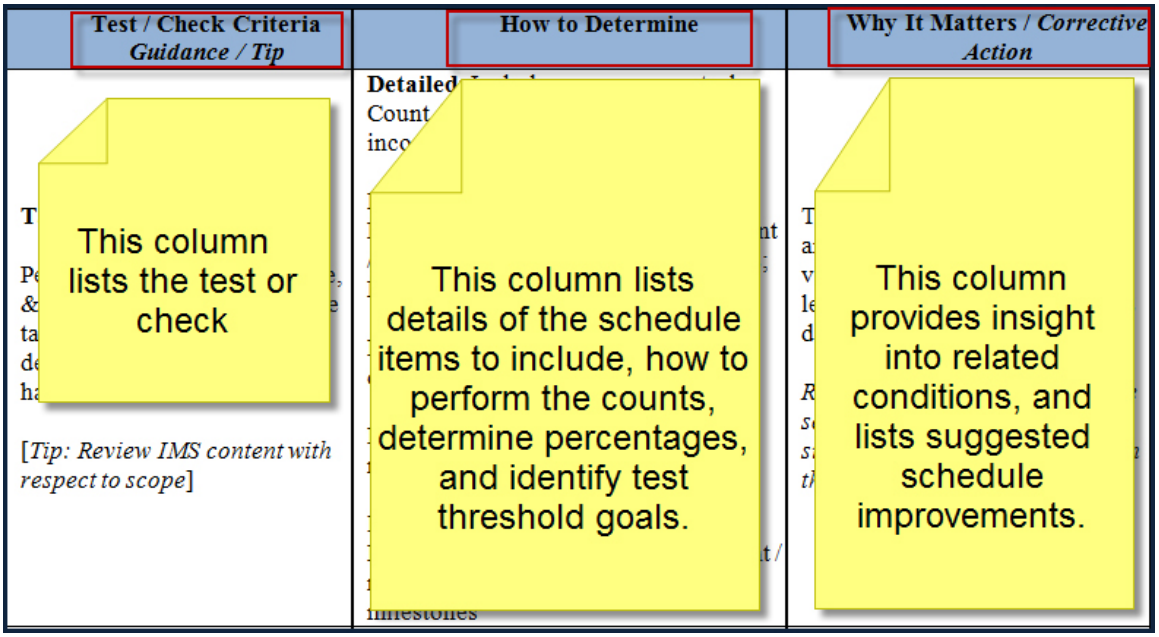
Introduction

Run!AzTech provides a series of automated filters aligned to the Generally Accepted Scheduling Principles (GASP)[1]. These principles are divided into Tenets with subsequent tests to support each. The twenty nine Quick Look tests featured in Run!AzTech help provide an automated answer to schedule validity. The first five tenets provide an answer to whether a schedule is Complete, Traceable, Transparent, Statused, & Predictive.

Run!AzTech GASP Filter Tables

For easy reference, the Quick Look Filters are arranged below. Columns explain each test by GASP tenet:

- The "**Test / Check Criteria; Guidance / Tip**" describes the "issue"
- The "**How to Determine**" describes the "test"
- The "**Why It Matters/ Corrective Action**" provides suggested schedule "fixes"



Use these tables along with Run!AzTech Quick Look filters to determine schedule alignment with the first five GASP tenets. Enjoy!

Note: Several GASP tests require in-depth analysis and automation provided in our comprehensive scheduling tool for MS Project, Run!AzTech. Other tests require a manual filtering approach, feel free to contact AzTech should you choose to purchase comprehensive support.

[1] Acknowledgement: The GASP concept comes from the National Defense Industrial Association (NDIA) Planning & Scheduling Excellence Guide (PASEG).

GASP Tenet 1: Complete

1. Complete - Schedules reflect comprehensive planning and are effective for execution. Level of Effort may be excluded from the IMS.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 1: Baseline Durations >2 Months Determine % of incomplete tasks with baseline durations greater than 44 working days (2 months). Tip: Shorter baseline durations reflect original planning scope granularity for efficient execution & precise performance measurement.	1. Apply Run!AzTech **Quick Look GASP 1: Test 1** 2. Observe & record percent score displayed in message box. Uses Quick Look Filters: QL_01A_BL_Dur_>2mo_Numerator QL_01B_BL_Dur_>2mo_Denominator Run!AzTech divides the numerator (N) count by the denominator (D) count. Goal: 5% or less. Compares: (N) number of incomplete, non-LOE, non-planning package, non-external, non-summary, non-milestone tasks that have baseline duration greater than 44 working days to (D) number of incomplete, non-LOE, non-planning package, non-external, non-summary, non-milestone tasks with baseline durations greater than 0 days.	Why It Matters: Shorter activities (2 months or less in duration) provide more visibility into how the activities are planned & allow a more objective progress evaluation. Corrective Action: Review & verify tasks with baseline durations longer than 44 working days or split into tasks less than 44 days.

1. Complete - Schedules reflect comprehensive planning and are effective for execution. Level of Effort may be excluded from the IMS.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 2: Forecast Durations > 2 Months Determine % of incomplete tasks with durations greater than 44 working days (2 months). Tip: Shorter task durations are easier to status & provide scope granularity for precise performance measurement.	1. Apply Run!AzTech **Quick Look GASP 1: Test 2** 2. Observe & record percent score displayed in message box. Uses Quick Look Filters: QL_02A_Fcst_Dur_>2mo_Numerator QL_02B_Fcst_Dur_>2mo _Denominator Run!AzTech divides the numerator (N) count by the denominator (D) count. Goal: 5% or less. Compares: (N) number of incomplete, non-LOE, non-planning package, non-external, non-summary, non-milestone tasks that have durations greater than 44 working days to (D) number of incomplete, non-LOE, non-planning package, non-external, non-summary, non-milestone tasks.	Why It Matters: Shorter tasks (2 months or less in duration) provide more visibility into how the tasks are planned & allow a more objective progress evaluation. Corrective Action: Review & verify tasks with forecast durations longer than 44 working days or split into tasks less than 44 days.

1. Complete - Schedules reflect comprehensive planning and are effective for execution. Level of Effort may be excluded from the IMS.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 3: Forecast Durations > 2 Months in 3 Month Look Ahead Determine % of incomplete tasks with durations greater than 44 working days (2 months) that are within next 3 months. Tasks clearly defined & well planned with easier to status shorter durations, provide granularity for precise performance measurement.	1. Apply Run!AzTech **Quick Look GASP 1: Test 3** 2. Observe & record percent score displayed in message box. Uses Quick Look Filters: QL_03A_Fcst_Dur_>2mo_within_3mo_Numerator QL_03B_Fcst_Dur_>2mo_within_3mo_Denominator Run!AzTech divides the numerator (N) count by the denominator (D) count. Goal: 5% or less. Compares: (N) number of incomplete, non-LOE, non-planning package, non-external, non-summary, non-milestone tasks activities within 3 months of status date that have durations greater than 44 working days to (D) number of incomplete, non-LOE, non-planning package, non-external, non-summary, non-milestone tasks within the same period.	Why It Matters: 3 month look ahead period scope must be understood & planned to execute efficiently. Shorter tasks (2 months or less in duration) provide more visibility into how the tasks are planned & allow a more objective progress evaluation. Corrective Action: Review & verify tasks with forecast durations longer than 44 working days or split into shorter tasks; apply this approach to advanced look ahead periods to affect changes.
Test 4: Estimated Durations Determine number of incomplete tasks with estimated durations. Tip: Indicates incomplete planning (durations have not been addressed).	1. Apply Run!AzTech **Quick Look GASP 1: Test 4** 2. Observe & record detected number displayed in message box. Uses Quick Look Filter: QL_04_Est_Dur Goal: Zero exceptions. Detects: number of incomplete tasks that have estimated durations.	Why It Matters: Estimated durations are the default in MSP indicating there has not been any duration input for that task. This suggests the planning has not yet been completed. Corrective Action: Replace estimated durations for all non-milestone tasks with durations from the CAM.

1. Complete - Schedules reflect comprehensive planning and are effective for execution. Level of Effort may be excluded from the IMS.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Tests 5 & 6: Missing Baseline Dates & Baseline Duration Determine all tasks without baseline dates & valid baseline durations. Tip: Cannot determine if tasks are early or late during execution without proper baseline.	1. Apply Run!AzTech **Quick Look GASP 1: Tests 5 & 6.** 2. Observe & record detected number displayed in each message box. Uses Quick Look Filters: QL_05_No_BL_Dates, then QL_06_No_BL_Dur Goal: All tasks have baseline dates & baseline duration. Detects: number of all tasks that do not have established baseline start, baseline finish, or baseline duration.	Why It Matters: Missing baseline information may indicate lapse in proper schedule management processes & exhibit lack of performance measure capabilities. Corrective Action: Populate & maintain proper baseline dates & durations (baseline the schedule).
Test 7: Cross Reference Fields Comprehensive data field referencing in IMS. Tip: Demonstrates source information tracks to each other, is represented in the IMS, & enables better program management.	Use the "**A_AllFields**" Table to identify related User Defined Fields for Test 7. Verify all documents cross-referenced to the IMS are represented with their own field in the IMS & are appropriately populated Required: CAMs, CAs, IMP, WBS, SOW, EVT, Work Package, Planning Package Recommended: OBS/IPT Determine related fields in the IMS for each artifact & search for completeness. Analyst uses judgment to determine if IMS is adequately cross-referenced. Goal: All required fields complete.	Why It Matters: Data cross reference fields exist & are populated to demonstrate source data alignment & provides a verifiable basis for IMS planning. Corrective Action: Populate & maintain proper artifact data fields in the IMS.

1. Complete - Schedules reflect comprehensive planning and are effective for execution. Level of Effort may be excluded from the IMS.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 8: Duplicate / Blank Names Search for blank or duplicate task names in the entire IMS. Tip: Unique & descriptive task names define the scope content & deliverable, aide user comprehension & facilitate determining progress during status.	Sort the entire IMS by task name, observe obvious task name duplicates & blank names for Test 8. Be aware of sorting parameters e.g. in MSP do not check the option *Keep Outline Structure* for sorting when including summary tasks; not checking the option eliminates outline structure as the primary sort that would prevent task name alignment as a primary sort for comparison. Through several iterations, search task names containing common words to discern repetitive phrases that do not exhibit uniqueness, such as several tasks that merely state "Perform Test", not differentiating specific tests. Goal: All names are unique & not blank.	Why It Matters: IMS task nomenclature is best understood when organized, unique, meaningful, & not reliant on summary or grouping titles to supplement their comprehension. Corrective Actions: Use present tense action verbs as described in the IMP if applicable, for each non-summary task where possible, when revising task names. Words such as analyze, design, draft, determine, produce, conduct, review & approve provide insight into unique descriptive task names & aid understanding each task deliverable.

GASP Complete Evaluation

GASP Tenet 2: Traceable

2. Traceable - Schedules have full network logic that reflects potential impacts to program completion. Schedules have populated code fields relating to required field mapping.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 9: Missing Logic Determine number of incomplete tasks without logic (predecessors or successors). Tip: Logic is fundamental for establishing an achievable schedule & imperative for its predictive capability. Missing logic calls into question schedule soundness & critical path validity.	1. Apply Run!AzTech **Quick Look GASP 2: Test 9** 2. Observe & record detected number displayed in message box. Uses Quick Look Filter: QL_09_No_Logic Goal: Zero exceptions. Detects: Number of incomplete, non-LOE, non-external, non-summary tasks that do not have at least one predecessor or one successor.	Why It Matters: External feed-in milestones w/o predecessor or feed-out milestones w/o successor may be appropriate, but all other activities need proper logic found within the IMS. Corrective Action: Determine appropriate predecessors & / or successors for tasks missing logic.
Test 10: Summary Logic (& Constraints / Deadlines) Identify summary tasks with applied logic or constraints. Tip: Applying logic or constraint to summary tasks potentially obscures impacts to detailed tasks & hinders schedule analysis.	1. Apply Run!AzTech **Quick Look GASP 2: Test 10** 2. Observe & record detected number displayed in message box. Uses Quick Look Filter: QL_10_Summary_Logic Goal: Zero exceptions. Detects: Number of all summary tasks that have predecessors or successors or constraint dates or deadline dates applied.	Why It Matters: Logic or constraints applied to summary tasks may have unintended consequences to subordinate detail tasks & may be difficult to discover when reviewing / analyzing schedule information. Corrective Action: Remove logic, constraints, & deadlines from summary tasks & apply logic & appropriate constraints & deadlines to detailed tasks.

2. Traceable - Schedules have full network logic that reflects potential impacts to program completion. Schedules have populated code fields relating to required field mapping.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 11: Finish-to-Start (FS) Relationships Determine % of incomplete tasks using FS relationships (preferred). Tip: FS relationships avoid scheduling activities in parallel & ensure the least opportunity for creating resource conflicts.	1. Apply Run!AzTech **Quick Look GASP 2: Test 11** 2. Observe & record percent score displayed in message box. Uses Quick Look Filters: QL_11A_FS_Rel_Numerator QL_11B_FS_Rel_Denominator Run!AzTech divides the numerator (N) count by the denominator (D) count. Goal: 90% or greater. Compares: (N) number of incomplete, non-LOE, non-summary tasks that have finish-to-start predecessor relationships to (D) number of incomplete, non-LOE, non-summary tasks.	Why It Matters: Promoting parallel activities risks scheduling more work than can be executed & potentially understates projecting accurate program finish. Corrective Action: Verify the use of any non-FS relationships & change to FS if appropriate.
Test 12: SS or SF Successor Without Either FS or FF Successor (Start-to-Start (SS) or Start-to-Finish (SF) Successor without either Finish-to-Start (FS) or Finish-to-Finish (FF) Successor) Determine number of incomplete activities using only SS or SF successor relationships. Tip: SS relationships may be valid, but not having at least one additional FS successor relationship prohibits establishing finish consequences, resulting in meaningless total float values.	1. Apply Run!AzTech **Quick Look GASP 2: Test 12** 2. Observe & record detected number displayed in message box. Run!AzTech function Goal: Zero exceptions. Detects: Number of incomplete, non-LOE tasks that have a SS or SF successor, but also do not have at least one FS or FF successor relationship to another task. Note: Condition, potentially equivalent of missing a successor.	Why It Matters: Relying only on SS or SF successor relationships does not model a finish consequence to the activity. Once in-progress, it loses its impact to other activities, does not retain priority to finishing & can reflect meaningless total float value to program end. Corrective Action: Determine & apply additional, appropriate FS or FF successor relationships.

2. Traceable - Schedules have full network logic that reflects potential impacts to program completion. Schedules have populated code fields relating to required field mapping.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 13: Total Float > 3 Months Determine % of tasks with total float >60 working days. Tip: Indicates a task may slip greater than 3 months without impact to program completion. Suggests a task is starting too early (missing an identified predecessor), or is not reflecting potential impacts to critical path (missing an identified successor). Possibility that some scope is not identified (tasks not present in the IMS).	1. Apply Run!AzTech **Quick Look GASP 2: Test 13** 2. Observe & record percent score displayed in message box. Uses Quick Look Filters: QL_13A_TF_>3mo_Numerator QL_13B_TF_>3mo_Denominator Run!AzTech divides the numerator (N) count by the denominator (D) count. Goal: 5% or less. Compares: (N) number of incomplete, non-LOE, non-summary tasks that have total float greater than 60 working days to (D) number of incomplete, non-LOE, non-summary tasks.	Why It Matters: Excessive total float is indication the task is not properly sequenced, either starting too early, or is missing a potential successor that could impact critical path determination & not properly forecasting program completion. Usually, identifying the end task in a path for missing successors is effective in addressing high total float for all tasks in the path. Corrective Action: Determine appropriate predecessors & / or successors for tasks with excessive total float. Tip: Sort the detected tasks in descending total float order to focus corrective actions on tasks with largest total float values.

2. Traceable - Schedules have full network logic that reflects potential impacts to program completion. Schedules have populated code fields relating to required field mapping.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 14: SNETs / FNETs Beyond 3 Month Look Ahead Determine % of SNET or FNET constraints on tasks > 3 month look ahead. Tip: Anticipate using less "no earlier than" constraints in periods further out, due to uncertainty & related rationale, relying more on logic alone to schedule a project.	1. Apply Run!AzTech **Quick Look GASP 2: Test 14** 2. Observe & record percent score displayed in message box. Uses Quick Look Filters: QL_14A_SNETorFNET_beyond _3mo_Numerator QL_14B_SNETorFNET_beyond _3mo_Denominator Run!AzTech divides the numerator (N) count by the denominator (D) count. Goal: 5% or less. Compares: (N) number of incomplete, non-LOE, non-summary, non-external tasks beyond 3 months from status date that have SNETs or FNETs to (D) number of incomplete, non-LOE, non-summary, non-external tasks beyond 3 months from status date.	Why It Matters: Generally, assumptions are less accurate in further look ahead periods, especially when attempting to model resource availability with SNETs / FNETs. Corrective Action: Review the "No Earlier Than" constraints & replace with logic relationships where practical.

2. Traceable - Schedules have full network logic that reflects potential impacts to program completion. Schedules have populated code fields relating to required field mapping.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 15: SNETs / FNETs within 3 Month Look Ahead Determine % of SNET or FNET constraints on tasks < =3 month look ahead. Tip: Anticipate using more "no earlier than" constraints in immediate period, due to certainty, to refine dates, where logic alone may not adequately model the project.	1. Apply Run!AzTech **Quick Look GASP 2: Test 15** 2. Observe & record percent score displayed in message box. Uses Quick Look Filters: QL_15A_SNETorFNET_within _3mo_Numerator QL_15B_SNETorFNET_within _3mo_Denominator Run!AzTech divides the numerator (N) count by the denominator (D) count. Goal: 10% or less. Compares: (N) number of incomplete, non-LOE, non-summary, non-external tasks within 3 months from status date that have SNETs or FNETs to (D) number of incomplete, non-LOE, non-summary, non-external tasks within 3 months from status date.	Why It Matters: Generally, conditions are well known in the immediate near term & predecessors alone may not sufficiently model resource availability for task execution. Use SNETs / FNETs appropriately, but not in place of logic. Corrective Action: Validate the "No Earlier Than" constraints & replace with logic relationships where practical.

GASP Traceable Evaluation

GASP Tenet 3: Transparent

3. Transparent - Schedules are constructed, used, maintained, and analyzed consistently with the IMS Supplemental Guidance (or equivalent documentation), rely on status and network logic as the primary forecast technique, identify risks and opportunities, and reflect rationale for constraints and lags.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 16: Tasks with Leads Determine number of incomplete tasks with leads > one day (imposed logic accelerations to successors). Tip: Ignores tasks finishing & successor starting on same day condition; Difficult to understand & manage "time overlap" created using leads.	1. Apply Run!AzTech **Quick Look GASP 3: Test 16** 2. Observe & record detected number displayed in message box. Uses Quick Look Filter: QL_16_Leads_>1d Note: Leads may be defined as a negative lag. Goal: Zero exceptions. Detects: Number of incomplete, non-LOE, non-summary tasks that have negative lag predecessors (greater than one day).	Why It Matters: Leads can distort total float & mask potential impacts to successor path tasks. Promote decomposing tasks & durations to facilitate Finish-to-Start relationships without leads. Corrective Action: Eliminate leads to allow schedule logic to drive dates.
Test 17: Tasks with Lags Determine % of incomplete tasks with lags (imposed logic delays to successors). Tip: Difficult to understand & manage "time gap" created using lags.	1. Apply Run!AzTech **Quick Look GASP 3: Test 17** 2. Observe & record percent score displayed in message box. Uses Quick Look Filters: QL_17A_Lags_Numerator QL_17B_Lags_Denominator Run!AzTech divides the numerator (N) count by the denominator (D) count. Goal: 5% or less. Compares: (N) number of incomplete, non-LOE, non-summary tasks that have predecessors with lag to (D) number of incomplete, non-LOE, non-summary tasks.	Why It Matters: Lags interject vagueness related to a "time gap" represented by the lag & are difficult to understand & manage. Lags should only model "wait time", not replace work effort or be used to anticipate successor start dates. Corrective Action: Minimize lags to allow schedule logic to drive dates.

3. Transparent - Schedules are constructed, used, maintained, and analyzed consistently with the IMS Supplemental Guidance (or equivalent documentation), rely on status and network logic as the primary forecast technique, identify risks and opportunities, and reflect rationale for constraints and lags.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 18: Constraints w/o Rationale Determine % of incomplete tasks that have constraints without comments (rationale) in Notes field. Note: Recognize that the schedule authors may utilize another custom field or document to explain constraints use (such as in the IMS Supplemental Guidance documentation), may need to adjust test results accordingly. Tip: Rationale aids understanding applied constraints.	1. Apply Run!AzTech **Quick Look GASP 3: Test 18** 2. Observe & record percent score displayed in message box. Uses Quick Look Filters: QL_18A_Constraints_No_Notes _Numerator QL_18B_Constraints_No_Notes _Denominator Run!AzTech divides the numerator (N) count by the denominator (D) count. Goal: 5% or less. Compares: (N) number of incomplete, non-LOE, non-summary tasks that are not ASAP & do not have Notes entries to (D) number of incomplete, non-LOE, non-summary tasks that are not ASAP.	Why It Matters: Documented explanations are required to understand constraint use, including validity & underlying intent. Aids in decision making & schedule maintenance. Corrective Action: Add explanations for deadlines & constraints to the Notes field.

3. Transparent - Schedules are constructed, used, maintained, and analyzed consistently with the IMS Supplemental Guidance (or equivalent documentation), rely on status and network logic as the primary forecast technique, identify risks and opportunities, and reflect rationale for constraints and lags.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 19: Lead/Lag w/o Rationale Determine % of incomplete tasks that have leads or lags without comments (rationale) in Notes field. Tip: Rationale aids understanding applied delays or accelerations.	1. Apply Run!AzTech **Quick Look GASP 3: Test 19** 2. Observe & record percent score displayed in message box. Uses Quick Look Filters: QL_19A_Leads_Lags_No_Notes _Numerator QL_19B_Leads_Lags_No_Notes _Denominator Run!AzTech divides the numerator (N) count by the denominator (D) count. Goal: 5% or less. Compares: (N) number of incomplete, non-LOE, non-summary tasks that have predecessor leads or lags & do not have note entries to (D) number of incomplete, non-LOE, non-summary tasks that have predecessor leads or lags.	Why It Matters: Rationale is required to understand lead / lag use, including validity & underlying intent. Aids in decision making & schedule maintenance. Corrective Action: Add explanations for leads / lags to the Notes field. Also see Leads (Test 16) above for alternative techniques.
Test 20: Hard Constraints Determine number of incomplete tasks utilizing hard constraints, prohibiting free flow of logic-driven IMS. Tip: Prevent dates from reflecting driving predecessor impacts. Includes: Must Start On Must Finish On Start No Later Than Finish No Later Than	1. Apply Run!AzTech **Quick Look GASP 3: Test 20** 2. Observe & record detected number displayed in message box. Uses Quick Look Filter: QL_20_Hard Constraints Goal: Zero exceptions. Detects: Number of incomplete, non-LOE tasks that have MSO or MFO or SNLT or FNLT constraints applied.	Why It Matters: Documented constraints affecting late dates may be necessary to establish key need dates & total float other than relying solely on backward pass calculations (use sparingly). Corrective Actions: Eliminate hard constraints from IMS & consider using deadlines instead. Deadlines enable forecast impacts while providing accurate total float values.

3. Transparent - Schedules are constructed, used, maintained, and analyzed consistently with the IMS Supplemental Guidance (or equivalent documentation), rely on status and network logic as the primary forecast technique, identify risks and opportunities, and reflect rationale for constraints and lags.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 21: Excessive Lags Determine number of incomplete tasks with excessive lags (delay values greater than one month). Tip: Excessive lag values potentially extend beyond one status period, complicating analysis of dates.	1. Apply Run!AzTech **Quick Look GASP 3: Test 21** 2. Observe & record detected number displayed in message box. Run!AzTech function Goal: Zero exceptions. Detects: Number of incomplete, non-LOE, non-summary tasks that have predecessors or successors with lag values greater than 20 working days.	Why It Matters: Excessive "wait time" complicates schedule management / visibility. Corrective Action: Replace excessive lags with documented / maintained "no earlier than" constraints".

GASP Transparent Evaluation

GASP Tenet 4: Statused

4. Statused - Schedules reflect valid actual and forecast dates, and tasks maintain previously established logical relationships.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 22: Invalid Forecast Dates Determine number of incomplete tasks that are not statused up to status date. Tip: Includes incomplete tasks without appropriate actual start or actual finish dates < status date, or in-progress tasks with remaining duration starting < status date. Tip: Unaccomplished work in the past is not accurate status, causes inaccurate projections, & diminishes schedule reliability.	1. Apply Run!AzTech **Quick Look GASP 4: Test 22** 2. Observe & record detected number displayed in message box. Uses Quick Look Filter: QL_22_Invalid_Forecast_Dates Goal: Zero exceptions Detects: Number of non-summary tasks that have forecast start or forecast finish dates earlier than the status date, without the applicable actual start or actual finish dates, or remaining duration not beginning at the status date for in-progress tasks. Note: IMS cannot have tasks with invalid forecast dates.	Why It Matters: It is not possible to perform future work in the past, therefore all tasks with work scheduled earlier than status date must re-schedule that work later than status date. Corrective Actions: Address invalid dates & incomplete tasks that are earlier than Timenow by providing accurate status & / or forecast dates. Not reflecting proper status jeopardizes performance measurement & successor path task projections.
Test 23: Invalid Actual Dates Determine number of tasks with actual start or actual finish dates in future. Tip: Tasks reflecting achievement in the future do not have accurate status; this causes inaccurate projections & diminishes schedule reliability.	1. Apply Run!AzTech **Quick Look GASP 4: Test 23** 2. Observe & record detected number displayed in message box. Uses Quick Look Filter: QL_23_Invalid_Actual_Dates Goal: Zero exceptions. Detects: Number of non-summary tasks that have actual start or actual finish dates later than the status date. Note: IMS cannot have tasks with invalid actual dates.	Why It Matters: Status date defines separation between past & future. It is not possible to accomplish effort in the future, beyond Timenow (status date). Corrective Actions: Correct the actual start or finish dates of tasks listed in the future. Not reflecting proper status jeopardizes performance measurement & successor path task projections.

4. Statused - Schedules reflect valid actual and forecast dates, and tasks maintain previously established logical relationships.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 24: Out-of-Sequence (OOS) Status Conditions Determine number of tasks that contain status conditions violating their logic relationships. Tip: Any tasks with out-of-sequence status condition render IMS projecting capabilities unreliable.	Review & detect tasks reflecting Actual Starts or Actual Finishes in current status cycle that are incongruent with predecessor logical relationships for Test 24. 1. Select a field to highlight all tasks within the IMS (e.g. Task Name) 2. Apply Run!AzTech **OOS** & follow prompts 3. Review the OOS report (Excel export) and address tasks highlighted with OOS status. E.g. an incomplete FS predecessor to an in-progress successor – that has an Actual Start & its predecessor does not have an Actual Finish, does not honor the relationship. Note: XLS report automatically saves to the same folder where the IMS is stored. Goal: Zero exceptions.	Why It Matters: Out-of-sequence status conditions override logic & potentially return overly optimistic successor path projections & meaningless total float values. Corrective Action: Resolve out-of-sequence status issues by either changing logic (if appropriate) or correcting the actual start or finish dates.

GASP Statused Evaluation

GASP Tenet 5: Predictive

5. Predictive - Schedules provide logic-driven forecast information, meaningful critical paths, and reflect achievable program completion dates.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 25: Push Forward Test Assess logic network integrity to program completion. Tip: Delaying an incomplete task with least total float reflects proportionate delay to program completion, demonstrating logic path to program completion.	Observe / record program completion milestone Early Finish date. Perform a successor trace by selecting a current period task with the least amount of total float, add 600 working days to existing duration, recalculate the schedule & click the *Trace* button**,** using the *R* (for Right) option. Verify the program completion milestone Early Finish date reflects a proportionate delay; the milestone is in the filtered set of tasks if it is logically tied to the successor trace task. Check for a logic break if the milestone is not present in the filtered set of tasks. Failed test when milestone does not reflect anticipated delay. Repeat this test on another current period task to ensure consistency. Note: If task with least total float has positive 25 working days total float, may only expect a 575 working day delaying impact to milestone.	Why It Matters: Adding 600 working days is more than two years duration, introducing dramatic impact to program completion. Failing the test indicates either broken logic exists or hard constraints prevent delays to successor path tasks. Corrective Action: Address missing logic or applied hard constraint issues.

Note: Please contact AzTech to purchase additional support for conducting manual schedule tests, or a comprehensive assessment of your system.

5. Predictive - Schedules provide logic-driven forecast information, meaningful critical paths, and reflect achievable program completion dates.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 26: Program Completion Trace Test Determine % of non-LOE, incomplete tasks logically tied to program completion. Tip: Feed-out tasks detected during this test should have documented rationale. Note any hard constraints assigned.	1. Perform a predecessor trace by selecting the program completion milestone & clicking the *Trace* button, using defaults (no options), highlight all tasks & Count. Note LOEs detected in path; decrement number of LOE from detected number for accurate calculation See Test 27 with respect to detected LOE. 2. Apply Run!AzTech **Quick Look GASP 5: Test 26** 3. Observe & record detected number displayed in message box for denominator. Uses Quick Look Filter: QL_26B_Program_Completion _Trace_Test_Denominator. Divide Trace Count by number of total incomplete, non-LOE, non-summary task (QL_26B). Goal: 95% & greater. Note: Review tasks not detected in the path by selecting Flag19 = “No” before continuing with other tests (Trace populates Flag19 with “Yes”). These are the tasks not logically tied to the program completion milestone.	Why It Matters: Although a percentage is calculated for test, it is more meaningful to review suspect tasks. Even a relatively few significant tasks without a successor path to program completion is reason for concern. Ideally all incomplete, non-LOE, non-summary tasks are logically tied to completion milestone. Corrective Actions: Investigate tasks not detected by the test, address missing successor path logic to milestone. Essential tasks not logically tied to program completion render IMS as not predictive & invalidate critical path.

5. Predictive - Schedules provide logic-driven forecast information, meaningful critical paths, and reflect achievable program completion dates.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 27: No LOE in Path to Program Completion Use the *Program Completion Trace Test* set-up for this check. Tip: Identify LOE tasks detected as having logical successor paths to program completion.	Perform a predecessor trace by selecting the program completion milestone & clicking the *Trace* button, using the *L* option to detect LOE in the path; window displays the number of LOE detected. Review the LOE tasks detected. Investigate to confirm that these LOE tasks are logically tied to discrete tasks & milestone & recommend changing logic. Goal: No LOE tied to discrete effort.	Why It Matters: LOE should not be logically tied to discrete work & should not be part of the critical path. Corrective Action: Investigate & remove LOE logic to discrete tasks & program completion to ensure LOE stays off of the critical path. Recommend using a LOE completion milestone to terminate LOE logic if necessary.
Test 28: Appropriate Constraints Applied to Endpoint Milestones Verify related milestones have appropriate constraints that provide meaningful schedule measures. Tip: Missing constraints diminish program management prioritization Avoid using hard constraints that override predictive nature of logic network.	Identify & review the endpoint milestones to ensure appropriate, documented constraints provide meaningful total float values & permit driving predecessors to establish forecast dates. Note: Method & rationale for establishing need dates (Late Dates) should align with IMS Supplemental Guidance documentation. Goal: All endpoint milestones should have constraints applied.	Why It Matters: Need dates reflect management's target. Constraints affecting the backward pass to program end & major milestones (if applicable) enable accurate total float calculation & permit precedence logic impacts. Corrective Action: Validate appropriate constraints used on endpoint milestones. Consider using documented deadlines.

5. Predictive - Schedules provide logic-driven forecast information, meaningful critical paths, and reflect achievable program completion dates.		
Test Description	**How to Determine**	**Why It Matters / Corrective Action**
Test 29: Critical Path Length Index (CPLI) Project performance indicates the ability to finish on time.	1. Determine working days duration from status date to program completion Early Finish date in IMS, **A** (critical path length). 2. Add amount of total float, **B** (least positive or negative value) to **A** & total. 3. Divide total (**A** + **B**) by **A** (critical path length, as determined above). **(A + B) / A** Goal: Should not be less than 0.95 with target of 1.00 (>1.00 is favorable <1.00 is unfavorable).	Why It Matters: Although geared towards performance, this test reflects IMS realism of completing on time & is meaningful when satisfactorily passing all previous GASP tests.

GASP Predictive Evaluation

Note: Please contact AzTech to purchase additional support for conducting manual schedule tests, or a comprehensive assessment of your system.

Key Web or E-Mail Links for More Info, Tech Support, or to Add to our Wish List

Run!AzTech Ribbon for MS Project - Check here for the latest on Run!AzTech Ribbon for MS Project 2013 and beyond and for Run!AzTech for MS Project 2010 and earlier.

http://goaztech.com/run-aztech-for-ms-project.aspx

AzTech Tool Suite Page- Check here for the latest on all of AzTech's tool suite offerings.

http://www.goAzTech.com/technology.aspx

AzTech Tool Suite Hotline - For technical assistance with any of our tools, please drop us a line at our Tool Suite Hotline.

http://www.goAzTech.com/aztech-tool-suite.aspx

AzTech Wish List - Have an idea or a suggestion on how to improve our tools? We love hearing from our customers, please shoot us an email!

wishlist@goAzTech.com

AzTech Tool FAQs – Check here to see if your question has already been answered!

http://www.goAzTech.com/faqs.aspx

