SCRUM Portfolio Planning and Delivery Optimization Model

Generated Assistant

August 11, 2025

Contents

1 1. Sets (Entities)

- $\mathcal{P}$: Projects (Entity: Project)

- $\mathcal{T}$: Teams (Entity: Team)

- $\mathcal{W}$: Workers (Entity: Worker)

- $\mathcal{F}$: Features (Entity: Feature)

- $\mathcal{S}$: Skills (Entity: Skill)

- $\mathcal{R}$: Roles (Entity: Role)

- $\mathcal{PO}$: Product Owners (Entity: ProductOwner)

- $\mathcal{SM}$: Scrum Masters (Entity: ScrumMaster)

- $\mathcal{PB}$: Product Backlogs (Entity: ProductBacklog)

- $\mathcal{SP}$: Sprints (Entity: Sprint)

- $\mathcal{SPP}$: Sprint Planning events (Entity: SprintPlanning)

- $\mathcal{DS}$: Daily Scrums (Entity: DailyScrum)

- $\mathcal{SR}$: Sprint Reviews (Entity: SprintReview)

- $\mathcal{SRE}$: Sprint Retrospectives (Entity: SprintRetrospective)

- $\mathcal{SBL}$: Sprint Backlogs (Entity: SprintBacklog)

- $\mathcal{SG}$: Sprint Goals (Entity: SprintGoal)

- $\mathcal{E}$: Epics (Entity: Epic)

- $\mathcal{US}$: User Stories (Entity: UserStory)

- $\mathcal{TSK}$: Tasks (Entity: Task)

- $\mathcal{DEV}$: Development Snapshots (Entity: DevelopmentSnapshot)

- $\mathcal{BL}$: Blockers (Entity: Blocker)

- $\mathcal{SH}$: Stakeholders (Entity: Stakeholder)

- $\mathcal{VEL}$: Velocity records (Entity: Velocity)

- $\mathcal{REP}$: Release Plans (Entity: ReleasePlan)

- $\mathcal{RM}$: Roadmaps (Entity: Roadmap)

- $\mathcal{SCB}$: Scrum Boards (Entity: ScrumBoard)

- $\mathcal{FED}$: Feature Documentation (Entity: FeatureDocumentation)

2   2. Indices

- $p \in \mathcal{P}$ (projects), $t \in \mathcal{T}$ (teams), $w \in \mathcal{W}$ (workers)

- $f \in \mathcal{F}$ (features), $e \in \mathcal{E}$ (epics), $u \in \mathcal{US}$ (user stories)

- $k \in \mathcal{TSK}$ (tasks), $bl \in \mathcal{BL}$ (blockers)

- $s \in \mathcal{SP}$ (sprints), $rep \in \mathcal{REP}$ (release plans)

- $vel \in \mathcal{VEL}$ (velocity records)

Parameters (from Entities/Attributes and Relationships)

- Feature attributes: $eff_f \geq 0$ (Feature.estimated_effort), $prio_f^F \geq 0$ (Feature.priority).

- User Story attributes: $prio_u^U \geq 0$ (UserStory.priority), $sp_u \geq 0$ (UserStory.story_points), $hasAcc_u \in \{0,1\}$ (UserStory.acceptance_criteria present).

- Task attributes: $eff_k^T \geq 0$ (Task.effort), $isCode_k \in \{0,1\}$ (Task.type = code), linkage $L_{k,u}^{US} \in \{0,1\}$ (Task $\rightarrow$ UserStory).

- Blocker attributes: $sev_{bl} \geq 0$ (Blocker.severity), linkage $L_{k,bl}^{BL} \in \{0,1\}$ (Task $\rightarrow$ Blocker).

- Velocity attributes: $\overline{v}_t$ (Velocity.avg._story_points for team t).

- Availability: $av_w \geq 0$ (Worker.availability); capacity transform α hours-to-story-points.

- Planning targets: $\widehat{cap}_s$ (target story points commitment for sprint s), DoD $\in \{0,1\}$ gate flag(s), allowed daily duration $\widehat{d}^{\text{daily}}$ (e.g. 15 minutes).

- Release dates: $date_{rep}^{plan}$ (ReleasePlan.planned_date), $date_{rep}^{calc}$ (calculated date from workload; model-dependent).

- Documentation links: $link_f^{FED} \in \{0,1\}$ (Feature has linked docs), att_s^{SR} (SprintReview.attendees_count).

Decision Variables (from DecisionVariables.csv)

- $x_{u,s} \in \{0,1\}$ (DV0: assign_user_story_to_sprint): story u is selected for sprint s.

- $y_{f,rep} \in \{0,1\}$ (DV1: select_feature_for_release): feature f included in release plan rep.

- $h_k \in \mathbb{Z}_{\geq 0}$ (DV2: allocate_task_effort): hours allocated to task k.

- $\hat{p}_u \in \{1,2,3,4,5\}$ (DV3: set_user_story_priority): planned execution priority for user story u during sprinting.

- $C_s \in \mathbb{Z}_{\geq 0}$ (DV4: team_capacity_committed): committed sprint capacity (story points) for sprint s.

- $r_{bl}^{BL} \in \{0,1\}$ (DV5: activate_blocker_resolution): 1 if blocker bl is scheduled for resolution in plan horizon.

- $a_{w,k} \in \{0,1\}$ (DV6: assign_worker_to_task): worker w works on task k.

- $q_u \in \mathbb{Z}_{\geq 0}$ (DV7: qa_test_allocation): number of test cases planned for user story u.

- $r_k^T \in \{0,1\}$ (DV8: code_review_required): task k has mandatory code review gate.

- $z_s \in \mathbb{Z}_{\geq 0}$ (DV9: bugfix_story_points_buffer): sprint s bugfix buffer (story points).

- $g \in \{0, 1, 2, 3, 4\}$ (DV10: ci_pipeline_gate_level): number of CI mandatory gates before merge (global or pipeline).

- $B_{rep} \in \{1, \ldots, 20\}$ (DV11: release_batch_size): number of features per release batch rep.

3   3. Goals

Generic multi-objective structure with scalarization (weights from Goals.csv):

$$\max / \min \; \sum_i w_i \cdot \text{Goal}_i \quad \text{with } w_i \text{ from CSV and signs given by GoalType.}$$

Below, each item shows ID, name, and mathematical form. (Weights shown as $W[\cdot]$.)

aximize_team_velocity

$$\max \; W[G0] \cdot \sum_{t \in \mathcal{T}} \overline{v}_t$$

(proxy objective to favor historically higher throughput teams in planning)

total_estimated_effort

$$\min \; W[G1] \cdot \sum_{f \in \mathcal{F}} \sum_{rep \in \mathcal{REP}} eff_f \, y_{f,rep}$$

ory_priority_coverage

$$\max \; W[G2] \cdot \sum_{u \in \mathcal{US}} \sum_{s \in \mathcal{SP}} prio_u^U \, x_{u,s}$$

open_blocker_severity

$$\min \; W[G3] \cdot \sum_{bl \in \mathcal{BL}} sev_{bl} \left(1 - r_{bl}^{BL}\right)$$

rint_goal_achievement

$$\max \; W[G4] \cdot \sum_{s \in \mathcal{SP}} \underbrace{\phi_s(C_s, \{x_{u,s}\}, z_s)}_{\text{proxy for goal achievement } \in [0,1]}$$

where ϕ_s is a monotone proxy increasing with capacity sufficiency vs. planned load.

nize_cycle_tasks_effort

$$\min \; W[G5] \cdot \sum_{k \in \mathcal{TSK}} h_k$$

imize_feature_priority

$$\max \; W[G6] \cdot \sum_{f \in \mathcal{F}} \sum_{rep \in \mathcal{REP}} prio_f^F \, y_{f,rep}$$

e_wip_status_pressure

$$\min \; W[G7] \cdot \sum_{k \in \mathcal{TSK}} \underbrace{\psi_k(\{a_{w,k}\})}_{\text{WIP proxy}}$$

with ψ_k increasing in parallel task starts / context switching.

nize_backlog_freshness

$$\max \; W[G8] \cdot \sum_{pb \in \mathcal{PB}} \text{freshness}(pb)$$

(data-quality proxy using last_updated; encourages grooming cadence)

...mize_release_slippage

$$\min \ W[G9] \cdot \sum_{rep \in \mathcal{REP}} \big| \, date_{rep}^{calc} - date_{rep}^{plan} \, \big|$$

...mize_team_utilization

$$\max \ W[G10] \cdot \frac{\sum_{w \in \mathcal{W}} \sum_{k \in \mathcal{TSK}} a_{w,k} \, h_k}{\sum_{w \in \mathcal{W}} av_w}$$

...int_duration_variance

$$\min \ W[G11] \cdot \sum_{s \in \mathcal{SP}} \big(\mathrm{duration}(s) - \widehat{\mathrm{duration}} \big)^2$$

4   4. Conditions

Each condition is expressed as a hard constraint ("must-match"), a soft constraint ("may-match") or a prohibition ("cannot-match") using the CSV *CriteriaType* = $\{2, 1, 0\}$ respectively. Below, we write the mathematical form and (where applicable) note the type.

...d_workers   (Must: 2)

$$\sum_{k \in \mathcal{TSK}} a_{w,k} \, h_k \ \leq \ av_w \qquad \forall w \in \mathcal{W}.$$

...out_story   (Must: 2)

$$\sum_{u \in \mathcal{US}} L_{k,u}^{US} \ \geq \ 1 \qquad \forall k \in \mathcal{TSK}.$$

...alignment   (Must: 2)

$$r_k^T \ \geq \ isCode_k \qquad \forall k \in \mathcal{TSK} \quad \text{(code tasks require review gate)}.$$

...ned_scope   (Must: 2)

$$\sum_{u \in \mathcal{US}} x_{u,s} \ \leq \ C_s + z_s \qquad \forall s \in \mathcal{SP}.$$

Interprets C_s (commitment) and z_s (bugfix buffer) as WIP cap in story-point units (via sp_u if desired).

..._coverage   (Must: 2)

$$x_{u,s} \ \leq \ hasAcc_u \qquad \forall u \in \mathcal{US}, \ \forall s \in \mathcal{SP}.$$

...ked_tasks   (Must: 2)

$$r_{bl}^{BL} \ \geq \ \min\big\{1, \sum_k L_{k,bl}^{BL}\big\} \qquad \forall bl \in \mathcal{BL}.$$

(If any task is blocked by bl, schedule its resolution.)

...wed_code   (Must: 2)

$$r_k^T \in \{0,1\}, \quad r_k^T = 1 \text{ if } isCode_k = 1.$$

(Equivalent to C2; kept explicit for audit.)

...rticipation   (May: 1)

$$att_s^{SR} \ \geq \ \widehat{A} \qquad \forall s \in \mathcal{SP}$$

where $\widehat{A}$ is a recommended threshold for Sprint Review attendees (soft target).

...ily_scrum   (Must: 2)

$$\mathrm{duration}(ds) \ \leq \ \widehat{d}^{\mathrm{daily}} \qquad \forall ds \in \mathcal{DS}.$$

e_quality   (Must: 2)

$$\mathbf{1}\{\text{documentation present at } spp\} \;=\; 1 \qquad \forall spp \in \mathcal{SPP}.$$

size_spikes   (May: 1)

$$\Delta \,\#(\text{entries in } pb) \;\leq\; \widehat{\Delta} \qquad \forall pb \in \mathcal{PB}.$$

n_linkage   (Must: 2)

$$link_f^{FED} \;=\; 1 \qquad \forall f \in \mathcal{F}.$$

5   5. DecisionVariables

n_user_story_to_sprint $x_{u,s} \in \{0,1\}$ for $u \in \mathcal{US}, s \in \mathcal{SP}$.

ect_feature_for_release $y_{f,rep} \in \{0,1\}$ for $f \in \mathcal{F}, rep \in \mathcal{REP}$.

2 allocate_task_effort $h_k \in \mathbb{Z}_{\geq 0}$ for $k \in \mathcal{TSK}$.

et_user_story_priority $\hat{p}_u \in \{1,2,3,4,5\}$ for $u \in \mathcal{US}$.

n_capacity_committed $C_s \in \mathbb{Z}_{\geq 0}$ for $s \in \mathcal{SP}$.

ate_blocker_resolution $r_{bl}^{BL} \in \{0,1\}$ for $bl \in \mathcal{BL}$.

assign_worker_to_task $a_{w,k} \in \{0,1\}$ for $w \in \mathcal{W}, k \in \mathcal{TSK}$.

V7 qa_test_allocation $q_u \in \mathbb{Z}_{\geq 0}$ for $u \in \mathcal{US}$.

code_review_required $r_k^T \in \{0,1\}$ for $k \in \mathcal{TSK}$.

ix_story_points_buffer $z_s \in \mathbb{Z}_{\geq 0}$ for $s \in \mathcal{SP}$.

ci_pipeline_gate_level $g \in \{0,1,2,3,4\}$.

11 release_batch_size $B_{rep} \in \{1,\ldots,20\}$ for $rep \in \mathcal{REP}$.

Additional Linking / Accounting Constraints (derived from Entities/Relations)

- Story-to-sprint workload accounting (optional):

$$\sum_{u \in \mathcal{US}} sp_u\, x_{u,s} \;\leq\; C_s + z_s \qquad \forall s \in \mathcal{SP}.$$

- Task assignment to availability (refines C0):

$$\sum_{k \in \mathcal{TSK}} a_{w,k}\, h_k \;\leq\; av_w \qquad \forall w \in \mathcal{W}.$$

- Release batch size control:

$$\sum_{f \in \mathcal{F}} y_{f,rep} \;\leq\; B_{rep} \qquad \forall rep \in \mathcal{REP}.$$