SCRUM Domain Optimization Model

Generated from Entities, Relationships, Goals, Conditions, Decision Variables

August 12, 2025

Contents

1 1. Sets (Entities)

- Project set $\mathcal{P}$ (Entities.csv: E0 `Project`, SetName `P`, Index p): product or initiative.
- Team set $\mathcal{T}$ (E1 `Team`, `T`, t): self-organized, cross-functional team.
- Worker set $\mathcal{W}$ (E2 `Worker`, `W`, w): individual team members.
- Feature set $\mathcal{F}$ (E3 `Feature`, `F`, f): mid-sized functionality.
- Skill set $\mathcal{S}$ (E4 `Skill`, `S`, s): worker competences.
- Role set $\mathcal{R}$ (E5 `Role`, `R`, r): responsibilities within Scrum team.
- Product Owner set $\mathcal{PO}$ (E6 `ProductOwner`, `PO`, po).
- Scrum Master set $\mathcal{SM}$ (E7 `ScrumMaster`, `SM`, sm).
- Product Backlog set $\mathcal{PB}$ (E8 `ProductBacklog`, `PB`, pb).
- Sprint set $\mathcal{SP}$ (E9 `Sprint`, `SP`, sp).
- Sprint Planning set $\mathcal{SPP}$ (E10 `SprintPlanning`, `SPP`, spp).
- Daily Scrum set $\mathcal{DS}$ (E11 `DailyScrum`, `DS`, ds).
- Sprint Review set $\mathcal{SR}$ (E12 `SprintReview`, `SR`, sr).
- Sprint Retrospective set $\mathcal{SRE}$ (E13 `SprintRetrospective`, `SRE`, sre).
- Sprint Backlog set $\mathcal{SBL}$ (E14 `SprintBacklog`, `SBL`, sbl).
- Sprint Goal set $\mathcal{SG}$ (E15 `SprintGoal`, `SG`, sg).
- Epic set $\mathcal{E}$ (E16 `Epic`, `E`, e).
- User Story set $\mathcal{US}$ (E17 `UserStory`, `US`, u).
- Task set $\mathcal{TSK}$ (E18 `Task`, `TSK`, k).
- Development Snapshot set $\mathcal{DEV}$ (E19 `DevelopmentSnapshot`, `DEV`, dv).
- Blocker set $\mathcal{BL}$ (E20 `Blocker`, `BL`, b).
- Stakeholder set $\mathcal{SH}$ (E21 `Stakeholder`, `SH`, h).
- Velocity set $\mathcal{VEL}$ (E22 `Velocity`, `VEL`, v).
- Release Plan set $\mathcal{REP}$ (E23 `ReleasePlan`, `REP`, rp).
- Roadmap set $\mathcal{RM}$ (E24 `Roadmap`, `RM`, rm).
- Scrum Board set $\mathcal{SCB}$ (E25 `ScrumBoard`, `SCB`, scb).
- Feature Documentation set $\mathcal{FED}$ (E26 `FeatureDocumentation`, `FED`, fd).
- **Relationship incidence sets (from Relationships.csv R1–R22):**
  - $A^{\text{proj}} \subseteq \mathcal{T} \times \mathcal{P}$ (R1 `is_assigned_to_project`)
  - $B^{\text{team}} \subseteq \mathcal{W} \times \mathcal{T}$ (R2 `belongs_to_team`)
  - $H^{\text{skill}} \subseteq \mathcal{W} \times \mathcal{S}$ (R3 `has_skill`)

- $T^{\text{role}} \subseteq \mathcal{W} \times \mathcal{R}$ (R4 `takes_on_role`)
- $M^{\text{backlog}} \subseteq \mathcal{PO} \times \mathcal{PB}$ (R5 `manages_backlog`)
- $U^{\text{support}} \subseteq \mathcal{T} \times \mathcal{SM}$ (R6 `is_supported_by`)
- $C^{\text{feat}} \subseteq \mathcal{PB} \times \mathcal{F}$ (R7 `contains_feature`)
- $C^{\text{epic}} \subseteq \mathcal{PB} \times \mathcal{E}$ (R8 `contains_epic`)
- $C^{\text{us}} \subseteq \mathcal{E} \times \mathcal{US}$ (R9 `contains_user_story`)
- $C^{\text{tsk}} \subseteq \mathcal{US} \times \mathcal{TSK}$ (R10 `consists_of_tasks`)
- $I^{\text{inSBL}} \subseteq \mathcal{US} \times \mathcal{SBL}$ (R11 `is_in_sprint_backlog`)
- $B^{\text{sbl\_sp}} \subseteq \mathcal{SBL} \times \mathcal{SP}$ (R12 `belongs_to_sprint`)
- $G^{\text{goal}} \subseteq \mathcal{SP} \times \mathcal{SG}$ (R13 `pursues_goal`)
- $Q^{\text{board}} \subseteq \mathcal{SCB} \times \mathcal{TSK}$ (R14 `contains_tasks`)
- $D^{\text{fed}} \subseteq \mathcal{FED} \times \mathcal{F}$ (R15 `documents_feature`)
- $X^{\text{blocked}} \subseteq \mathcal{TSK} \times \mathcal{BL}$ (R16 `is_blocked_by`)
- $P^{\text{review}} \subseteq \mathcal{SH} \times \mathcal{SR}$ (R17 `participates_in`)
- $R^{\text{retromod}} \subseteq \mathcal{SM} \times \mathcal{SRE}$ (R18 `moderates_retrospective`)
- $V^{\text{team}} \subseteq \mathcal{VEL} \times \mathcal{T}$ (R19 `refers_to_team`)
- $L^{\text{release}} \subseteq \mathcal{REP} \times \mathcal{F}$ (R20 `plans_release`)
- $Z^{\text{road}} \subseteq \mathcal{REP} \times \mathcal{RM}$ (R21 `is_part_of_roadmap`)
- $Y^{\text{snap}} \subseteq \mathcal{SP} \times \mathcal{DEV}$ (R22 `generates_snapshot`)

2 2. Indices

- $p \in \mathcal{P}$, $t \in \mathcal{T}$, $w \in \mathcal{W}$, $f \in \mathcal{F}$, $s \in \mathcal{S}$, $r \in \mathcal{R}$, $po \in \mathcal{PO}$, $sm \in \mathcal{SM}$, $pb \in \mathcal{PB}$, $sp \in \mathcal{SP}$, $spp \in \mathcal{SPP}$, $ds \in \mathcal{DS}$, $sr \in \mathcal{SR}$, $sre \in \mathcal{SRE}$, $sbl \in \mathcal{SBL}$, $sg \in \mathcal{SG}$, $e \in \mathcal{E}$, $u \in \mathcal{US}$, $k \in \mathcal{TSK}$, $dv \in \mathcal{DEV}$, $b \in \mathcal{BL}$, $h \in \mathcal{SH}$, $v \in \mathcal{VEL}$, $rp \in \mathcal{REP}$, $rm \in \mathcal{RM}$, $scb \in \mathcal{SCB}$, $fd \in \mathcal{FED}$.

Attribute parameters (from Entities.csv). For any entity X and its attribute `attr`, we denote the parameter value as $\text{attr}(x)$, e.g., $\text{budget}(p)$, $\text{team\_size}(t)$, $\text{story\_points}(u)$, $\text{effort}(k)$, $\text{max\_velocity}(t)$, etc. Non-numeric attributes are handled via indicator parameters, e.g., $\text{isActiveStatus}(sp) \in \{0,1\}$, $\text{isCrossFunctional}(t) \in \{0,1\}$, $\text{hasFeedback}(sr) \in \{0,1\}$, and so on.

Relationship helpers. Write $C^{\text{tsk}}(u)$ for the set $\{k \in \mathcal{TSK} : (u,k) \in C^{\text{tsk}}\}$, and similarly for other incidence sets.

3 3. Goals

- **G0** `maximize_team_velocity`
  Logic: Maximize recorded maximum velocity across teams.
  Math:
  $$\max \sum_{t \in \mathcal{T}} 1.0 \cdot \text{max\_velocity}(t)$$
- **G1** `minimize_open_blocker_severity`
  Logic: Minimize the sum of severities of open blockers.
  Math:
  $$\min \sum_{b \in \mathcal{BL}} 1.0 \cdot \text{severity}(b) \cdot \text{isOpen}(b)$$

- **G2** `minimize_total_task_effort`
  Logic: Minimize planned effort over all tasks.
  Math:

$$\min \sum_{k \in \mathcal{TSK}} 1.0 \cdot \text{eff\_plan}(k) \quad \text{with eff\_plan}(k) = \text{DV8 set\_task\_effort}(k)$$

- **G3** `maximize_story_points_completed`
  Logic: Maximize delivered story points of user stories placed into sprints.
  Math:

$$\max \sum_{u \in \mathcal{US}} \sum_{sp \in \mathcal{SP}} 1.0 \cdot \text{pts}(u) \cdot y_{u,sp} \quad \text{with pts}(u) = \text{DV7 set\_story\_points}(u),\ y_{u,sp} = \text{DV1}$$

- **G4** `maximize_sprint_goal_achievement`
  Logic: Maximize achievement targets for all sprints.
  Math:

$$\max \sum_{sp \in \mathcal{SP}} 1.0 \cdot g_{sp} \quad \text{with } g_{sp} = \text{DV5 set\_sprint\_goal\_target}(sp)$$

- **G5** `minimize_feature_estimated_effort`
  Logic: Minimize total estimated effort of selected features.
  Math:

$$\min \sum_{f \in \mathcal{F}} 1.0 \cdot \text{estimated\_effort}(f) \cdot x_f \quad \text{with } x_f = \text{DV0 select\_feature}(f)$$

- **G6** `minimize_project_budget_usage`
  Logic: Minimize allocated budget across projects.
  Math:

$$\min \sum_{p \in \mathcal{P}} 1.0 \cdot b_p \quad \text{with } b_p = \text{DV3 allocate\_budget\_to\_project}(p)$$

- **G7** `maximize_team_satisfaction`
  Logic: Maximize retrospective team satisfaction.
  Math:

$$\max \sum_{sre \in \mathcal{SRE}} 1.0 \cdot \text{team\_satisfaction}(sre)$$

- **G8** `maximize_stakeholder_relevance`
  Logic: Maximize relevance of participating stakeholders (e.g., those linked to reviews).
  Math:

$$\max \sum_{(h,sr) \in P^{\text{review}}} 1.0 \cdot \text{relevance\_to\_feature}(h)$$

- **G9** `minimize_sprint_backlog_total_effort`
  Logic: Minimize total effort capacity assigned to sprint backlogs.
  Math:

$$\min \sum_{sbl \in \mathcal{SBL}} 1.0 \cdot c_{sbl} \quad \text{with } c_{sbl} = \text{DV12 set\_sprint\_backlog\_effort\_cap}(sbl)$$

- **G10** `maximize_review_attendance`
  Logic: Maximize attendance targets for sprint reviews.
  Math:

$$\max \sum_{sr \in \mathcal{SR}} 1.0 \cdot a_{sr} \quad \text{with } a_{sr} = \text{DV10 set\_review\_attendance\_target}(sr)$$

- **G11** `minimize_product_backlog_entries`
  Logic: Minimize number of items in all product backlogs.
  Math:
$$\min \sum_{pb \in \mathcal{PB}} 1.0 \cdot \text{number\_of\_entries}(pb)$$

4 4. Conditions

- **C0** `team_size_within_limit`
  Logic: Team size must be within allowed bounds.
  Math:
$$3 \leq s_t \leq 15 \quad \forall t \in \mathcal{T} \quad \text{with } s_t = \text{DV4 set\_team\_size}(t)$$

- **C1** `team_type_is_cross_functional`
  Logic: Selected teams must be cross-functional.
  Math:
$$\text{isCrossFunctional}(t) = 1 \quad \forall t \in \mathcal{T}$$

- **C2** `worker_availability_above_threshold`
  Logic: Workers must meet minimum availability to be assigned tasks.
  Math:
$$\sum_{k \in \mathcal{TSK}} a_{k,w} \leq \text{availability}(w) \quad \forall w \in \mathcal{W}, \quad a_{k,w} = \text{DV2 assign\_task\_to\_worker}(k, w)$$

- **C3** `sprint_status_is_planned_or_active`
  Logic: Only planned or active sprints are eligible to receive work.
  Math:
$$\text{isPlannedOrActive}(sp) = 1 \;\Rightarrow\; \sum_{u \in \mathcal{US}} y_{u,sp} \geq 0, \quad \text{isPlannedOrActive}(sp) = 0 \;\Rightarrow\; y_{u,sp} = 0 \; \forall u$$

- **C4** `user_story_status_not_done`
  Logic: Stories marked done are not reconsidered/assigned.
  Math:
$$\text{isDone}(u) = 1 \;\Rightarrow\; \sum_{sp \in \mathcal{SP}} y_{u,sp} = 0 \quad \forall u \in \mathcal{US}$$

- **C5** `task_type_is_valid`
  Logic: Tasks must be of allowed types (e.g., development, test).
  Math:
$$\text{isAllowedType}(k) = 1 \quad \forall k \in \mathcal{TSK}$$

- **C6** `blocker_status_must_be_resolved`
  Logic: Any blocker linked to a completed task must be resolved and under the maximum allowed severity.
  Math:
$$\text{isDone}(k) = 1 \;\wedge\; (k, b) \in X^{\text{blocked}} \;\Rightarrow\; \text{isResolved}(b) = 1$$
$$\text{severity}(b) \leq \text{sev\_max} \quad \forall b \in \mathcal{BL}, \quad \text{sev\_max} = \text{DV9}$$

- **C7** `feature_priority_meets_minimum`
  Logic: Selected features must meet a minimum priority.
  Math:
$$x_f = 1 \;\Rightarrow\; \text{priority}(f) \geq \pi_{\min} \quad \forall f \in \mathcal{F},\; x_f = \text{DV0},\; \pi_{\min} \in \{1, \ldots, 5\}$$

- **C8** `project_budget_under_cap`
  Logic: Allocated budget may not exceed the project budget cap.
  Math:
  $$0 \leq b_p \leq \text{budget}(p) \quad \forall p \in \mathcal{P}, \quad b_p = \text{DV3}$$

- **C9** `sprint_goal_must_be_defined`
  Logic: Each sprint has a defined (non-empty) objective.
  Math:
  $$(sp, sg) \in G^{\text{goal}} \;\Rightarrow\; \text{isDefinedObjective}(sg) = 1 \quad \forall sp \in \mathcal{SP}$$

- **C10** `review_feedback_documented`
  Logic: Sprint reviews must include feedback documentation.
  Math:
  $$\text{hasFeedback}(sr) = 1 \quad \forall sr \in \mathcal{SR}$$

- **C11** `retrospective_has_moderation`
  Logic: Retrospectives are moderated (by some Scrum Master).
  Math:
  $$\exists\, sm \in \mathcal{SM} :\; (sm, sre) \in R^{\text{retromod}} \quad \forall sre \in \mathcal{SRE}$$

- **C12** `velocity_min_threshold`
  Logic: Each team must meet a minimum velocity threshold.
  Math:
  $$\text{max\_velocity}(t) \;\geq\; \nu_t^{\min} \quad \forall t \in \mathcal{T}, \qquad \nu_t^{\min} = \text{DV11 set\_min\_velocity\_threshold}(t)$$

- **Linking/consistency constraints (using relationships).**
  - If a story is assigned to a sprint, then that story must be in the sprint's backlog:
    $$y_{u,sp} = 1 \Rightarrow \exists\, sbl \in \mathcal{SBL} :\; (u, sbl) \in I^{\text{inSBL}} \wedge (sbl, sp) \in B^{\text{sbl\_sp}}$$
  - Effort capacity cap per sprint backlog:
    $$\sum_{u:(u,sbl)\in I^{\text{inSBL}}} \sum_{k \in C^{\text{tsk}}(u)} \text{eff\_plan}(k) \;\leq\; c_{sbl} \quad \forall sbl \in \mathcal{SBL}, \quad c_{sbl} = \text{DV12}$$
  - Each task assigned to a worker belongs to some story:
    $$a_{k,w} = 1 \Rightarrow \exists\, u \in \mathcal{US} :\; (u, k) \in C^{\text{tsk}} \qquad \forall k, w$$

5 5. DecisionVariables

- **DV0** `select_feature`: $x_f \in \{0, 1\}$ for each $f \in \mathcal{F}$ (Binary). Domain $\{0, 1\}$, Min 0, Max 1.
- **DV1** `assign_user_story_to_sprint`: $y_{u,sp} \in \{0, 1\}$ for each $(u, sp) \in \mathcal{US} \times \mathcal{SP}$ (Binary). Domain $\{0, 1\}$, Min 0, Max 1.
- **DV2** `assign_task_to_worker`: $a_{k,w} \in \{0, 1\}$ for each $(k, w) \in \mathcal{TSK} \times \mathcal{W}$ (Binary). Domain $\{0, 1\}$, Min 0, Max 1.
- **DV3** `allocate_budget_to_project`: $b_p \in \mathbb{R}_{\geq 0}$ for $p \in \mathcal{P}$. Domain $\mathbb{R}$, Min 0, Max 10^6 (cap).

- **DV4** `set_team_size`: $s_t \in \mathbb{Z}$ for $t \in \mathcal{T}$. Domain $\mathbb{Z}$, Min 3, Max 15.
- **DV5** `set_sprint_goal_target`: $g_{sp} \in \mathbb{Z}$ for $sp \in \mathcal{SP}$. Domain $\mathbb{Z}$, Min 0, Max 100.
- **DV6** `prioritize_feature`: $\mathrm{pr}_f \in \mathbb{Z}$ for $f \in \mathcal{F}$. Domain $\mathbb{Z}$, Min 1, Max 5.
- **DV7** `set_story_points`: $\mathrm{pts}(u) \in \mathbb{Z}$ for $u \in \mathcal{US}$. Domain $\mathbb{Z}$, Min 1, Max 13.
- **DV8** `set_task_effort`: $\mathrm{eff\_plan}(k) \in \mathbb{Z}$ for $k \in \mathcal{TSK}$. Domain $\mathbb{Z}$, Min 1, Max 16.
- **DV9** `set_max_blocker_severity_allowed`: $\mathrm{sev\_max} \in \mathbb{Z}$. Domain $\mathbb{Z}$, Min 0, Max 10.
- **DV10** `set_review_attendance_target`: $a_{sr} \in \mathbb{Z}$ for $sr \in \mathcal{SR}$. Domain $\mathbb{Z}$, Min 0, Max 200.
- **DV11** `set_min_velocity_threshold`: $\nu_t^{\min} \in \mathbb{Z}$ for $t \in \mathcal{T}$. Domain $\mathbb{Z}$, Min 0, Max 200.
- **DV12** `set_sprint_backlog_effort_cap`: $c_{sbl} \in \mathbb{Z}$ for $sbl \in \mathcal{SBL}$. Domain $\mathbb{Z}$, Min 0, Max 1000.