SCRUM Domain Optimization Model

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

August 12, 2025

Contents

1 1. Sets (Entities) 2

2 2. Indices 3

3 3. Goals 4

4 4. Conditions 5

5 5. DecisionVariables 7

1 1. Sets (Entities)

- $\mathcal{P}$: Projects (`Project`)
- $\mathcal{T}$: Teams (`Team`)
- $\mathcal{W}$: Workers (`Worker`)
- $\mathcal{F}$: Features (`Feature`)
- $\mathcal{SK}$: Skills (`Skill`)
- $\mathcal{RO}$: Roles (`Role`)
- $\mathcal{PO}$: Product Owners (`ProductOwner`)
- $\mathcal{SM}$: Scrum Masters (`ScrumMaster`)
- $\mathcal{PB}$: Product Backlogs (`ProductBacklog`)
- $\mathcal{SP}$: Sprints (`Sprint`)
- $\mathcal{SPP}$: Sprint Plannings (`SprintPlanning`)
- $\mathcal{DS}$: Daily Scrums (`DailyScrum`)
- $\mathcal{SR}$: Sprint Reviews (`SprintReview`)
- $\mathcal{SRE}$: Sprint Retrospectives (`SprintRetrospective`)
- $\mathcal{SBL}$: Sprint Backlogs (`SprintBacklog`)
- $\mathcal{SG}$: Sprint Goals (`SprintGoal`)
- $\mathcal{E}$: Epics (`Epic`)
- $\mathcal{US}$: User Stories (`UserStory`)
- $\mathcal{TSK}$: Tasks (`Task`)
- $\mathcal{DEV}$: Development Snapshots (`DevelopmentSnapshot`)
- $\mathcal{BL}$: Blockers (`Blocker`)
- $\mathcal{SH}$: Stakeholders (`Stakeholder`)
- $\mathcal{VEL}$: Velocity records (`Velocity`)
- $\mathcal{REP}$: Release Plans (`ReleasePlan`)
- $\mathcal{RM}$: Roadmaps (`Roadmap`)
- $\mathcal{SCB}$: Scrum Boards (`ScrumBoard`)
- $\mathcal{FED}$: Feature Docs (`FeatureDocumentation`)

Relationship-induced index sets (from `Relationships.csv`). We use the following incidence sets:

- $\mathcal{T}(p) \subseteq \mathcal{T}$: teams assigned to project p (R1).
- $\mathcal{W}(t) \subseteq \mathcal{W}$: workers belonging to team t (R2).
- $\mathcal{SK}(w) \subseteq \mathcal{SK}$: skills of worker w (R3).
- $\mathcal{RO}(w) \subseteq \mathcal{RO}$: roles of worker w (R4).
- $\mathcal{F}(pb) \subseteq \mathcal{F}$: features contained in product backlog pb (R7).
- $\mathcal{E}(pb) \subseteq \mathcal{E}$: epics in product backlog (R8).
- $\mathcal{US}(e) \subseteq \mathcal{US}$: user stories in epic e (R9).
- $\mathcal{TSK}(us) \subseteq \mathcal{TSK}$: tasks in user story us (R10).
- $\mathcal{US}(\,sbl\,) \subseteq \mathcal{US}$: stories assigned to sprint backlog sbl (R11).
- $sbl(s) \in \mathcal{SBL}$: unique sprint backlog of sprint s (R12).
- $sg(s) \in \mathcal{SG}$: goal of sprint s (R13).
- $\mathcal{TSK}(scb) \subseteq \mathcal{TSK}$: tasks on scrum board scb (R14).
- $f(FED) \in \mathcal{F}$: feature documented by a feature doc (R15).
- $\mathcal{BL}(tsk) \subseteq \mathcal{BL}$: blockers of task tsk (R16).
- $\mathcal{SR}(sh) \subseteq \mathcal{SR}$: reviews a stakeholder attends (R17).
- $\mathcal{SRE}(sm) \subseteq \mathcal{SRE}$: retros moderated by scrum master sm (R18).
- $t(vel) \in \mathcal{T}$: team referred to by velocity record (R19).
- $\mathcal{F}(rep) \subseteq \mathcal{F}$: features planned in release plan (R20).
- $rm(rep) \in \mathcal{RM}$: roadmap of release plan (R21).
- $dev(s) \in \mathcal{DEV}$: snapshot generated by sprint s (R22).

2 2. Indices

- $p \in \mathcal{P}$, $t \in \mathcal{T}$, $w \in \mathcal{W}$, $f \in \mathcal{F}$, $sk \in \mathcal{SK}$, $ro \in \mathcal{RO}$
- $po \in \mathcal{PO}$, $sm \in \mathcal{SM}$, $pb \in \mathcal{PB}$, $s \in \mathcal{SP}$, $spp \in \mathcal{SPP}$
- $ds \in \mathcal{DS}$, $sr \in \mathcal{SR}$, $sre \in \mathcal{SRE}$, $sbl \in \mathcal{SBL}$, $sg \in \mathcal{SG}$
- $e \in \mathcal{E}$, $us \in \mathcal{US}$, $tsk \in \mathcal{TSK}$, $dev \in \mathcal{DEV}$, $bl \in \mathcal{BL}$
- $sh \in \mathcal{SH}$, $vel \in \mathcal{VEL}$, $rep \in \mathcal{REP}$, $rm \in \mathcal{RM}$, $scb \in \mathcal{SCB}$

Selected parameters (from attributes). We use numeric parameters derived from entity attributes:

- $budget_p \geq 0$ (`Project.budget`); $status_p^{\text{active}} \in \{0,1\}$ (`Project.status=active`)
- $avail_w \in [0,1]$ (`Worker.availability`); $status_w^{\text{active}} \in \{0,1\}$
- $teamstat_t^{\text{available}} \in \{0,1\}$; $size^{\min}, size^{\max}$ bounds for team size
- $prio_f \in \mathbb{Z}_+$; $\widehat{eff}_f \geq 0$ (`Feature.estimated_effort`)
- $sev_{bl} \in \mathbb{Z}_+$; $resolved_{bl} \in \{0,1\}$
- $att_{sr} \in \mathbb{Z}_+$ (`SprintReview.attendees_count`)
- $goalAch_s \in [0,1]$ (`Sprint.achievement_of_goal`)
- $vel_{vel}^{avg} \geq 0$ and mapping $t(vel)$ (`Velocity.avg_story_points`)
- $totEff_{sbl} \geq 0$, $nTasks_{sbl} \in \mathbb{Z}_+$ (`SprintBacklog`)
- $rel_{sh} \in [0,1]$ (`Stakeholder.relevance_to_feature`)

3 3. Goals

We maximize a weighted sum of *maximize_...* terms and minimize a weighted sum of *minimize_...* terms by converting minimization components to maximization (via subtraction from upper bounds or negation). Let $\lambda_g > 0$ denote the CSV weight of goal g and Norm$[\cdot]$ a positive scaling to place heterogeneous terms on comparable ranges.

- **G0 maximize_avg_story_points** (Weight 1.0, CriteriaType 1)
  Logic: Prefer teams with higher historical velocity.
  Math: $\max\ \lambda_{G0} \sum_{vel \in \mathcal{VEL}} \text{Norm}\big[vel_{vel}^{avg}\big]$
- **G1 minimize_blocker_severity** (Weight 1.0, CriteriaType 2)
  Logic: Reduce unresolved blocker impact.
  Math: $\min\ \lambda_{G1} \sum_{tsk \in \mathcal{TSK}} \sum_{bl \in \mathcal{BL}(tsk)} (1 - resolved_{bl})\ sev_{bl}$
- **G2 minimize_task_effort** (Weight 0.9, CriteriaType 1)
  Logic: Prefer selecting/assigning lower-effort tasks. Let $a_{tsk,w} \in \{0,1\}$ (DV4).
  Math: $\min\ \lambda_{G2} \sum_{tsk} \sum_{w} a_{tsk,w}\ effort_{tsk}$
- **G3 maximize_sprint_goal_achievement** (Weight 1.0, CriteriaType 2)
  Logic: Favor sprints with higher goal achievement.
  Math: $\max\ \lambda_{G3} \sum_{s \in \mathcal{SP}} \text{Norm}\big[goalAch_s\big]$
- **G4 minimize_feature_estimated_effort** (Weight 0.8, CriteriaType 1)
  Logic: Prefer lower-effort features when selected for a sprint. Let $y_{f,s} \in \{0,1\}$ (DV2 adapted over sprints).
  Math: $\min\ \lambda_{G4} \sum_{s} \sum_{f} y_{f,s}\ \widehat{eff}_f$

- **G5 maximize_stakeholder_relevance** (Weight 0.7, CriteriaType 1)
  Logic: Prefer items valuable to highly relevant stakeholders. Map features to stakeholders by participation in reviews of sprints that include those features.
  Math (proxy): max $\lambda_{G5} \sum_{s} \sum_{f} y_{f,s} \Big(\sum_{sr \in \mathcal{SR}} \sum_{sh \in \mathcal{SH}} \mathbf{1}[sr \in \mathcal{SR}(sh)]\ rel_{sh} \Big)$

- **G6 minimize_total_sprint_effort** (Weight 1.0, CriteriaType 2)
  Logic: Keep planned effort within capacity.
  Math: min $\lambda_{G6} \sum_{s} totEff_{sbl(s)}$

- **G7 maximize_attendees_in_review** (Weight 0.5, CriteriaType 1)
  Logic: Prefer sprints with broader review participation.
  Math: max $\lambda_{G7} \sum_{sr \in \mathcal{SR}} \text{Norm}[att_{sr}]$

- **G8 minimize_number_of_tasks** (Weight 0.6, CriteriaType 1)
  Logic: Prefer smaller WIP.
  Math: min $\lambda_{G8} \sum_{s} nTasks_{sbl(s)}$

- **G9 maximize_worker_availability** (Weight 1.0, CriteriaType 2)
  Logic: Assign work to more-available workers.
  Math: max $\lambda_{G9} \sum_{tsk} \sum_{w} a_{tsk,w}\ \text{Norm}[avail_w]$

- **G10 minimize_project_budget** (Weight 0.9, CriteriaType 1)
  Logic: Prefer plans consuming less budget. If $b_f \geq 0$ is allocated feature budget (DV8), assign features to projects by backlog.
  Math: min $\lambda_{G10} \sum_{p \in \mathcal{P}} \sum_{pb:\, pb \text{ of } p} \sum_{f \in \mathcal{F}(pb)} b_f \leq \sum_{p} budget_p$ (also enforced as constraint in Sec. 4)

- **G11 maximize_scrum_master_experience** (Weight 0.4, CriteriaType 1)
  Logic: Prefer assigning more-experienced SMs. Let $z_{sm,t} \in \{0,1\}$ (DV5). Parameter $exp_{sm} \geq 0$.
  Math: max $\lambda_{G11} \sum_{t \in \mathcal{T}} \sum_{sm \in \mathcal{SM}} z_{sm,t}\ \text{Norm}[exp_{sm}]$

Composite objective. With $\mathcal{G}_{\max}$ the set of maximize-goals and $\mathcal{G}_{\min}$ minimize-goals (converted by sign/upper bounds), we solve:

$$\max \sum_{g \in \mathcal{G}_{\max}} \lambda_g \cdot \text{Term}_g \ - \sum_{g \in \mathcal{G}_{\min}} \lambda_g \cdot \text{Term}_g^{(\min)}$$

subject to the conditions in Section 4 and variable domains in Section 5.

4 4. Conditions

Interpretation rule: For each condition Ck, CriteriaType = 2 (*Must-Match*) is a hard constraint; = 1 (*May-Match*) is modeled as a soft preference added to the objective (positive bonus or negative penalty); = 0 (*Cannot-Match*) forbids assignments/selections.

Let further decision variables from Sec. 5 be available. We list each condition with logic and mathematics.

- **C0 active_projects_only** (Must, IsSum=False)
  Logic: Only active projects participate.
  Math: $\forall p \in \mathcal{P}:\ status_p^{\text{active}} = 1$ (or equivalently, filter $\mathcal{P} \leftarrow \{p : status_p^{\text{active}} = 1\}$).

- **C1 team_status_available** (Must)
  Logic: Only available teams can be used.
  Math: $\forall t \in \mathcal{T}:\ teamstat_t^{\text{available}} = 1$.

- **C2 worker_status_active** (Must)
  Logic: Only active workers can be assigned.
  Math: $\forall w \in \mathcal{W}, \forall t:\ x_{w,t} \le status_w^{\text{active}},\ \forall p:\ u_{w,p} \le status_w^{\text{active}}$.

- **C3 min_worker_availability** (Must)
  Logic: Assignments require $avail_w \ge \alpha$ (threshold).
  Math: $\forall w, t:\ x_{w,t} \le \mathbf{1}[avail_w \ge \alpha]$, and $\forall w, tsk:\ a_{tsk,w} \le \mathbf{1}[avail_w \ge \alpha]$.

- **C4 user_story_ready** (Must)
  Logic: Only *ready* user stories may be selected into a sprint. Let $\texttt{ready}_{us} \in \{0,1\}$.
  Math: $\forall s, us:\ y_{us,s}^{US} \le \texttt{ready}_{us}$.

- **C5 feature_priority_threshold** (Must)
  Logic: Selected features must have $prio_f \ge \underline{\pi}$.
  Math: $\forall s, f:\ y_{f,s} \le \mathbf{1}[\,_f \ge \underline{\pi}\,]$.

- **C6 sprint_in_progress_or_planned** (Must)
  Logic: Consider only sprints with admissible status. Let $\texttt{ok}_s \in \{0,1\}$.
  Math: $\forall s:\ \texttt{ok}_s = 1$ (or restrict $\mathcal{SP}$ accordingly).

- **C7 exclude_unresolved_blockers** (Cannot-Match, IsSum=True)
  Logic: Tasks with unresolved blockers cannot be assigned.
  Math: $\forall tsk, \forall bl \in \mathcal{BL}(tsk):\ (1 - resolved_{bl}) = 1 \Rightarrow \sum_w a_{tsk,w} = 0$.

- **C8 prefer_dev_task_type** (May-Match)
  Logic: Prefer `Task.type=development`. Let $\texttt{dev}_{tsk} \in \{0,1\}$. Weight 0.5.
  Math (objective bonus): $+\,0.5 \sum_{tsk,w} a_{tsk,w} \cdot \texttt{dev}_{tsk}$.

- **C9 backlog_status_refined** (Must)
  Logic: Only refined product backlogs contribute features/stories. Let $\texttt{refined}_{pb} \in \{0,1\}$.
  Math: $\forall pb:\ \texttt{refined}_{pb} = 1$ and for $f \in \mathcal{F}(pb)$ or $us \in \mathcal{US}(e), e \in \mathcal{E}(pb)$, selections must satisfy $y_{f,s} \le \texttt{refined}_{pb}$ and $y_{us,s}^{US} \le \texttt{refined}_{pb}$.

- **C10 release_plan_planned** (Must)
  Logic: Only planned release plans are considered. Let $\texttt{planned}_{rep} \in \{0,1\}$.
  Math: $\forall rep:\ \texttt{planned}_{rep} = 1$.

- **C11 roadmap_end_date_defined** (Must)
  Logic: Roadmaps must have end date set. Let $\texttt{hasEnd}_{rm} \in \{0,1\}$.
  Math: $\forall rep:\ \texttt{hasEnd}_{rm(rep)} = 1$.

Capacity, budget, and linking (implied by entities/relationships).

- *Worker–Team assignment (DV0):* $x_{w,t} \in \{0,1\}$ and $\sum_t x_{w,t} \le 1$.

- *Worker–Project assignment (DV1):* $u_{w,p} \in \{0,1\}$ and $u_{w,p} \le \sum_{t \in \mathcal{T}(p)} x_{w,t}$ (project via team).

- *ScrumMaster–Team (DV5):* $z_{sm,t} \in \{0,1\}$ and $\sum_{sm} z_{sm,t} = 1$.
- *ProductOwner–Project (DV6):* $q_{po,p} \in \{0,1\}$ and $\sum_{po} q_{po,p} = 1$.
- *Feature–Sprint selection (DV2):* $y_{f,s} \in \{0,1\}$; if $f \in \mathcal{F}(pb)$ associated to project p, then $y_{f,s} \leq \sum_{po} q_{po,p}$.
- *Story–Sprint selection (DV3):* $y^{US}_{us,s} \in \{0,1\}$ with $us \in \mathcal{US}(e)$ and (optionally) $y^{US}_{us,s} \leq \sum_{f \in \mathcal{F}(pb)} y_{f,s}$ if stories are gated by chosen features/epics.
- *Task assignment (DV4):* $a_{tsk,w} \in \{0,1\}$ and $\sum_{w} a_{tsk,w} \leq 1$; link to stories $tsk \in \mathcal{TSK}(us)$ implies $a_{tsk,w} \leq \sum_{s} y^{US}_{us,s}$.
- *Sprint effort vs availability:* $\sum_{us} y^{US}_{us,s} \cdot \mathtt{SP}_{us} \leq \sum_{w} cap_{w,s}$ where $\mathtt{SP}_{us}$ are (DV11) story points and $cap_{w,s} = avail_w \cdot Cap^{\max}$.
- *Budget (DV8, G10):* $\sum_f b_f \leq \sum_p budget_p$ and $b_f \leq M \sum_s y_{f,s}$.
- *Team size (DV9):* $size^{\min} \leq n_t \leq size^{\max}$ with $n_t = \sum_w x_{w,t}$.
- *Sprint length (DV7):* $7 \leq L_s \leq 30$ (days).
- *Feature priority (DV10):* $1 \leq \pi_f \leq 5$ and if $y_{f,s} = 1$ then $\pi_f \geq \underline{\pi}$ (duplicate of C5).

5 5. DecisionVariables

Let the following decision variables correspond exactly to `DecisionVariables.csv` (domains and bounds included).

- **DV0 assign_worker_to_team**: $x_{w,t} \in \{0,1\}$, $w \in \mathcal{W}, t \in \mathcal{T}$ (1 if worker w is assigned to team t).
- **DV1 assign_worker_to_project**: $u_{w,p} \in \{0,1\}$, $w \in \mathcal{W}, p \in \mathcal{P}$ (1 if worker w contributes to project p).
- **DV2 select_feature_for_sprint**: $y_{f,s} \in \{0,1\}$, $f \in \mathcal{F}, s \in \mathcal{SP}$ (1 if feature f is in sprint s).
- **DV3 select_user_story_for_sprint**: $y^{US}_{us,s} \in \{0,1\}$, $us \in \mathcal{US}, s \in \mathcal{SP}$.
- **DV4 assign_task_to_worker**: $a_{tsk,w} \in \{0,1\}$, $tsk \in \mathcal{TSK}, w \in \mathcal{W}$.
- **DV5 choose_scrum_master_for_team**: $z_{sm,t} \in \{0,1\}$, $sm \in \mathcal{SM}, t \in \mathcal{T}$.
- **DV6 choose_product_owner_for_project**: $q_{po,p} \in \{0,1\}$, $po \in \mathcal{PO}, p \in \mathcal{P}$.
- **DV7 set_sprint_length_days**: $L_s \in \mathbb{Z}$, $s \in \mathcal{SP}$ with $7 \leq L_s \leq 30$.
- **DV8 allocate_budget_to_feature**: $b_f \in \mathbb{R}_+$, $f \in \mathcal{F}$ with $0 \leq b_f \leq 1{,}000{,}000$.
- **DV9 set_team_size**: $n_t \in \mathbb{Z}$, $t \in \mathcal{T}$ with $3 \leq n_t \leq 20$.
- **DV10 set_feature_priority**: $\pi_f \in \mathbb{Z}$, $f \in \mathcal{F}$ with $1 \leq \pi_f \leq 5$.
- **DV11 decide_story_points_for_user_story**: $\mathtt{SP}_{us} \in \mathbb{Z}$, $us \in \mathcal{US}$ with $1 \leq \mathtt{SP}_{us} \leq 13$.