APS Developer Cheat Sheet

Single-page reference for Autodesk Platform Services APIs

RAPS Development Team

January 2026

Contents

1   APS Developer Cheat Sheet

- Single-page reference for Autodesk Platform Services (APS) with RAPS CLI shortcuts**

- –

1.1   Authentication

1.1.1   OAuth Flow Comparison

2-Legged OAuth	3-Legged OAuth
Use: Server-to-server	**Use:** User-facing apps
Context: App identity	**Context:** User + app
BIM360: Limited access	**BIM360:** Full access
Grant: `client_credentials`	**Grant:** `authorization_code`
RAPS: `raps auth login`	**RAPS:** `raps auth login --3legged`

1.1.2 Essential OAuth Scopes

Scope	Purpose	RAPS Usage
`data:read`	Read files/projects	`--scopes data:read`
`data:write`	Modify metadata	`--scopes data:read,data:write`
`data:create`	Upload files	`--scopes data:read,data:write,data:create`
`bucket:read`	List buckets	`--scopes bucket:read`
`bucket:create`	Create buckets	`--scopes bucket:read,bucket:create`
`viewables:read`	View models	`--scopes viewables:read`
`code:all`	Design Automation	`--scopes code:all`
`account:read`	BIM360/ACC info	`--scopes account:read`

- –

1.2 Data Management API

1.2.1 Common Endpoints

Operation	Manual API Call	RAPS Command
List Hubs	`GET /project/v1/hubs`	`raps dm hubs`
List Projects	`GET /project/v1/hubs/{id}/projects`	`raps dm projects`
List Folders	`GET /data/v1/projects/{id}/folders/{id}/contents`	`raps dm folders <project>`
Upload File	Multi-step: Create storage → Upload → Create item → Create version	`raps dm upload <file> --project <id>`
Download	`GET /data/v1/projects/{id}/downloads` + download URL	`raps dm download <item_id>`

1.2.2 URN Formats

Context	URN Format	Example
OSS Object	`urn:adsk.objects:os.object:{bucket}:{obj}`	`urn:adsk.objects:os.object:mybucket:mo`
Data Mgmt	`urn:adsk.wipprod:dm.lineage:{item_id}`	`urn:adsk.wipprod:dm.lineage:abc123...`
Encoded	Base64 URL-safe encoding	`dXJuOmFkc2sub2JqZWN0cy...`

- URN Encoding:**

```
# Manual (error-prone)
echo -n "urn:..." | base64 | tr '+/' '-_' | tr -d '='

# With RAPS
raps urn encode "urn:adsk.objects:os.object:bucket/file.dwg"
```

- –

1.3 Object Storage Service (OSS)

1.3.1 Bucket Operations

Operation	Manual API	RAPS Command
List Buckets	`GET /oss/v2/buckets`	`raps bucket list`
Create Bucket	`POST /oss/v2/buckets`	`raps bucket create <name>`
Delete Bucket	`DELETE /oss/v2/buckets/{key}`	`raps bucket delete <name>`

1.3.2 Object Operations

Operation	Manual API	RAPS Command
List Objects	GET /oss/v2/buckets/{key}/objects	raps oss list <bucket>
Upload Object	PUT /oss/v2/buckets/{key}/objects/{name}	raps oss upload <file> <bucket>
Download	GET /oss/v2/buckets/{key}/objects/{name}	raps oss download <bucket> <object>

1.3.3 Bucket Naming Rules

- 3-128 characters, lowercase, numbers, hyphens

- Must start/end with letter or number

- No spaces, underscores, IP addresses

- −

1.4 Model Derivative API

1.4.1 Translation Workflow

```
1. Upload → 2. Start Translation → 3. Poll Status → 4. Download Results
```

1.4.2 Translation Operations

Operation	Manual Process	RAPS Command
Start Translation	POST /modelderivative/v2/designdata/job	raps translate <urn> --formats svf2
Check Status	GET /modelderivative/v2/designdata/{urn}/manifest	raps translate status <urn>
Wait Complete	Manual polling with delays	raps translate <urn> --wait
Get Properties	GET /modelderivative/v2/designdata/{urn}/metadata/{guid}/properties	raps translate properties <urn>

1.4.3 Output Formats

Format	Use Case	RAPS Code
SVF2	Web viewer (modern)	svf2
SVF	Web viewer (legacy)	svf
OBJ	3D meshes	obj
STL	3D printing	stl
PDF	2D drawings	pdf
DWG	AutoCAD format	dwg

- −

1.5 Design Automation API

1.5.1 Engine Support

Engine	File Types	RAPS Command
AutoCAD	DWG, DXF	raps da engines --filter autocad
Revit	RVT, RFA	raps da engines --filter revit
Inventor	IPT, IAM	raps da engines --filter inventor
3ds Max	MAX	raps da engines --filter max

1.5.2 DA Components

```
AppBundle (code) + Activity (process) + WorkItem (execution)
```

Component	RAPS Management
AppBundle	`raps da appbundle create`
Activity	`raps da activity create`
WorkItem	`raps da workitem submit`

- –

1.6 Common Error Codes

Code	Cause	RAPS Solution
400	Invalid request format	Built-in validation
401	Invalid/expired token	`raps auth refresh`
403	Insufficient permissions	`raps auth status --scopes`
404	URN not found	`raps urn encode`
409	Bucket name conflict	`raps bucket list` first
429	Rate limit exceeded	Built-in rate limiting

- –

1.7 Rate Limits

API	Limit	RAPS Handling
Authentication	500/min	Automatic token reuse
Data Management	100/min	Intelligent batching
Model Derivative	20 concurrent	Queue management
OSS	500/min	Parallel optimization

- –

1.8 Quick Setup Commands

1.8.1 First-Time Setup

```
# Install RAPS
# Windows: scoop install raps
# macOS: brew install raps
# Linux: cargo install raps-cli

# Authenticate
raps auth login

# Test
raps auth status
```

1.8.2 Common Workflows

```
# Upload → Translate → View
raps oss upload model.rvt mybucket
raps translate $(raps urn encode "urn:adsk.objects:os.object:mybucket:model.rvt") --wait
raps view $(raps urn encode "urn:adsk.objects:os.object:mybucket:model.rvt")

# Batch Operations
raps oss upload-batch *.dwg --bucket mybucket
raps translate-batch --bucket mybucket --formats svf2
```

- –

1.9 Emergency Commands

```
# Authentication issues
raps auth refresh
raps auth login --force
raps auth status --verbose

# Operation failures
raps health check --comprehensive
raps logs --level error --last 1h
```

- –

1.10 Pro Tips

- Use `raps examples` for common workflows

- Set up profiles: `raps auth login --profile <name>`

- Enable verbose: `--verbose` flag for debugging

- Health check: `raps health check` for issues

- –

- Support:** RAPS Discord | GitHub

- Resources:** rapscli.xyz | APS Docs

- –

- Print-friendly format • Save for reference • Share with team*

- RAPS v4.2.1 | APS APIs: Auth v2, DM v1, MD v2, OSS v2, DA v3 | January 2026*