

RAPS Developer Quick Start Guide

Get started with RAPS CLI for APS automation

RAPS Development Team

January 2026

Table of Contents

1 RAPS Developer Quick Start Cheat Sheet	1
1.1 Installation & Setup (< 5 minutes)	2
1.1.1 1. Install RAPS	2
1.1.2 2. Authenticate	2
1.1.3 3. Configure Profile	2
1.2 Essential Commands	2
1.2.1 Authentication & Config	2
1.2.2 Object Storage Service (OSS)	2
1.2.3 Data Management (DM)	3
1.2.4 Model Derivative	3
1.2.5 Bulk Operations	3
1.3 Common Workflows	3
1.3.1 Quick File Upload	3
1.3.2 Project Setup	3
1.3.3 CI/CD Integration	4
1.3.4 Monitoring & Debugging	4
1.4 Power User Tips	4
1.4.1 Performance Optimization	4
1.4.2 Advanced Configuration	4
1.4.3 Security Best Practices	4
1.5 Troubleshooting	5
1.5.1 Common Issues & Solutions	5
1.5.2 Debug Mode	5
1.5.3 Getting Help	5
1.6 Environment Variables	5
1.7 Integration Examples	6
1.7.1 GitHub Actions	6
1.7.2 Docker	6
1.7.3 Python Integration	6
1.8 Quick Reference Card	6
1.8.1 Most Used Commands	6
1.8.2 Emergency Commands	6
1.9 Version Compatibility Notes	6
1.9.1 RAPS 4.2.1 Changes	6
1.9.2 Backward Compatibility	7
1.9.3 Upgrade Path	7

1 RAPS Developer Quick Start Cheat Sheet

- —
- Document Version**: v4.2 (January 2026)
- RAPS Version**: 4.2.1
- APS API Compatibility**: Data Management v1, Model Derivative v2, OSS v2, Authentication v2

- Minimum Rust Version**: 1.88.0
- —

1.1 Installation & Setup (< 5 minutes)

1.1.1 1. Install RAPS

```
# Windows (Scoop)
scoop bucket add raps https://github.com/raps-cli/scoop-bucket
scoop install raps

# macOS (Homebrew)
brew tap raps-cli/tap
brew install raps

# Linux (Cargo)
cargo install raps-cli

# Or download from https://rapscli.xyz/download
```

1.1.2 2. Authenticate

```
# Interactive login (recommended)
raps auth login

# Or set credentials
raps auth set --client-id YOUR_CLIENT_ID --client-secret YOUR_SECRET

# Verify authentication
raps auth status
```

1.1.3 3. Configure Profile

```
# Set default profile
raps config set profile default

# Create environment-specific profiles
raps auth login --profile production
raps auth login --profile staging
```

- —

1.2 Essential Commands

1.2.1 Authentication & Config

Command	Description
raps auth login	Interactive OAuth login
raps auth status	Check authentication status
raps auth refresh	Refresh expired tokens
raps config list	Show all configuration
raps config set profile <name>	Switch active profile

1.2.2 Object Storage Service (OSS)

Command	Description
raps oss buckets	List all buckets
raps oss create-bucket <name>	Create new bucket

Command	Description
raps oss upload <file> <bucket>	Upload single file
raps oss upload-batch --folder <dir>	Upload entire directory
raps oss download <urn> <path>	Download file
raps oss list <bucket>	List bucket contents

1.2.3 Data Management (DM)

Command	Description
raps dm projects	List all projects
raps dm folders <project-id>	List project folders
raps dm upload <file> <project>	Upload to project
raps dm versions <item-id>	Show item versions
raps dm create-folder <name>	Create folder

1.2.4 Model Derivative

Command	Description
raps derivative start <urn>	Start translation job
raps derivative status <urn>	Check job status
raps derivative formats <urn>	List available formats
raps derivative download <urn>	Download derivatives

1.2.5 Bulk Operations

Command	Description
raps batch upload <pattern>	Upload matching files
raps batch process <list>	Process file list
raps batch status	Monitor batch jobs
raps batch retry <job-id>	Retry failed operations

• —

1.3 Common Workflows

1.3.1 Quick File Upload

```
# Upload and process in one command
raps upload-and-process model.dwg --project my-project --formats svf,pdf

# With progress monitoring
raps upload model.rvt --project design --watch-progress

# Batch upload with parallel processing
raps batch upload *.dwg --project construction --parallel 5
```

1.3.2 Project Setup

```
# Create complete project structure
raps dm create-project "New Building" \
  - folders "Models,Drawings,Documents" \
  - permissions team-read

# Setup development environment
raps config create-workspace dev \
```

```
- project-id 12345 \
- default-bucket dev-models \
- auto-derivatives
```

1.3.3 CI/CD Integration

```
# In your GitHub Actions / Jenkins:
raps auth login --token $APS_TOKEN
raps deploy --environment production \
- source ./models \
- auto-rollback \
- notify-slack $SLACK_WEBHOOK
```

1.3.4 Monitoring & Debugging

```
# Check system status
raps health check

# View detailed logs
raps logs --level debug --follow

# Performance monitoring
raps stats --operations upload --timeframe 24h
```

• —

1.4 Power User Tips

1.4.1 Performance Optimization

```
# Optimal parallel uploads (adjust based on network)
raps config set parallel.uploads 10

# Enable compression for large files
raps config set compression.enabled true

# Use regional endpoints for better speed
raps config set region us-west
```

1.4.2 Advanced Configuration

```
# Custom retry logic
raps config set retry.max-attempts 5
raps config set retry.backoff exponential

# Webhook notifications
raps webhook add success https://api.yourapp.com/aps-success
raps webhook add error https://api.yourapp.com/aps-error

# Custom output formats
raps config set output.format json # json, yaml, table, csv
```

1.4.3 Security Best Practices

```
# Use environment variables (recommended)
export APS_CLIENT_ID="your-client-id"
export APS_CLIENT_SECRET="your-secret"

# Enable audit logging
```

```
raps config set audit.enabled true
raps config set audit.destination ./logs/

# Rotate credentials regularly
raps auth rotate --schedule monthly
```

• —

1.5 Troubleshooting

1.5.1 Common Issues & Solutions

Issue	Solution
Authentication failed	Run <code>raps auth refresh</code> or <code>raps auth login</code>
File upload timeout	Increase timeout: <code>raps config set timeout 300</code>
Rate limit exceeded	Add delays: <code>raps config set rate-limit.delay 1000</code>
Permission denied	Check APS scopes: <code>raps auth status --verbose</code>
Network errors	Use retry: <code>raps config set retry.network true</code>

1.5.2 Debug Mode

```
# Enable verbose output
raps --verbose <command>

# Save debug information
raps debug info > debug-info.txt

# Test connectivity
raps health test-connection
```

1.5.3 Getting Help

```
# Built-in help
raps help # General help
raps help upload # Command-specific help
raps --help # CLI options

# Examples and tutorials
raps examples # Show common examples
raps tutorial interactive # Interactive tutorial
```

• —

1.6 Environment Variables

Variable	Description	Example
APS_CLIENT_ID	Application client ID	your-client-id
APS_CLIENT_SECRET	Application secret	your-secret
APS_CALLBACK_URL	OAuth callback	http://localhost:8080/callback
RAPS_PROFILE	Default profile	production
RAPS_CONFIG_PATH	Config file location	~/.config/raps/
RAPS_LOG_LEVEL	Logging verbosity	debug, info, warn, error

• —

1.7 Integration Examples

1.7.1 GitHub Actions

```
- name: Deploy to APS
  run: |
    raps auth login --token ${{ secrets.APS_TOKEN }}
    raps batch upload ./models --project ${{ env.PROJECT_ID }}
```

1.7.2 Docker

```
FROM rust:alpine
RUN cargo install raps-cli
COPY . /app
WORKDIR /app
CMD ["raps", "batch", "process", "queue.json"]
```

1.7.3 Python Integration

```
import subprocess

# Call RAPS from Python
result = subprocess.run([
    'raps', 'upload', 'model.dwg',
    '--project', project_id,
    '--output', 'json'
], capture_output=True, text=True)

response = json.loads(result.stdout)
```

- —

1.8 Quick Reference Card

1.8.1 Most Used Commands

raps auth login	<i># Authenticate</i>
raps upload file.dwg --project p1	<i># Upload file</i>
raps dm projects	<i># List projects</i>
raps batch upload *.rvt	<i># Bulk upload</i>
raps derivative start <urn>	<i># Process model</i>
raps health check	<i># System status</i>

1.8.2 Emergency Commands

raps auth refresh	<i># Fix auth issues</i>
raps batch cancel <job-id>	<i># Stop runaway job</i>
raps logs --level error	<i># Check errors</i>
raps config reset	<i># Reset configuration</i>

- Full Documentation**: rapscli.xyz/docs
- Support**: support@rapscli.xyz
- Community**: discord.gg/raps
- —

1.9 Version Compatibility Notes

1.9.1 RAPS 4.2.1 Changes

- Enhanced MCP integration for natural language operations

- Improved parallel processing performance (10% faster)
- New `raps health monitor` command for real-time monitoring
- Extended APS API timeout handling

1.9.2 Backward Compatibility

- RAPS 4.x**: All commands compatible
- RAPS 3.x**: Core commands compatible, some advanced features unavailable
- APS API Changes**: Model Derivative v2 required for new formats

1.9.3 Upgrade Path

```
# Check current version
raps --version

# Upgrade to latest
# macOS: brew upgrade raps
# Windows: scoop update raps
# Linux: cargo install --force raps-cli
```

- —
- RAPS Developer Cheat Sheet v4.2 | RAPS v4.2.1 | APS APIs: DM v1, MD v2, OSS v2, Auth v2 | Updated: January 2026*