

```

    }
}

Asset aReturn = Asset.Load(szFolderPath);

return aReturn.IsLoaded ? aReturn : null;
}

public static string GetAllRelatedAssetIds(string szAssetId, ref string szAssetIdString, string szSitePath)
{
    string szRet = "";

    if (!string.IsNullOrEmpty(szAssetId) && !string.Equals(szAssetId, "0", StringComparison.OrdinalIgnoreCase) && !int.Equals(szAssetId, 0))
    {
        if (!szAssetId.Contains("|" + szAssetId + "|"))
        {
            szAssetIdString = string.IsNullOrEmpty(szAssetIdString) ? "|" + szAssetId + "|" : szAssetIdString + szAssetId + "|";

            FilterParams fpFilter = new FilterParams();
            fpFilter.Add(AssetPropertyNames.TemplateLabel, Comparison.Equals, "Relationship Config");
            fpFilter.Add("filter_string", Comparison.Equals, "|" + szAssetId + "|");

            List<Asset> lpRelationshipConfig = Asset.Load(szSitePath + "/Relationships Config/").GetFilterList(fpFilter);
            foreach (Asset aData in lpRelationshipConfig)
                szRet = string.Equals(szAssetId, aData["source_id"], StringComparison.OrdinalIgnoreCase) ? aData["destination_id"] : aData["source_id"];
        }
    }

    return szRet;
}

public static IEnumerable<LocaleId> CreateLocaleConfigCache(string szSitePath)
{
    return Asset.Load(szSitePath + "/Locales Config/").GetFileList().Select(aData => new LocaleId(aData));
}

public static string GetLocaleId(Asset asset, string szSitePath)
{
    string szRet = "";

    List<Asset> laLocalConfig = Asset.Load(szSitePath + "/Locales Config/").GetFileList();
    foreach (Asset aData in laLocalConfig)
    {
        string szAssetPath = asset.AssetId.ToString().ToUpper();
        if (asset.Type.Equals(AssetType.Folder))
            szAssetPath = asset.AssetId.ToString().ToUpper() + (string.Equals(szAssetPath.Last(), "/" ? "" : "/"));

        if (szAssetPath.Contains(aData["folder_root"].ToUpper() + (string.Equals(aData["folder_root"].Last(), "/" ? "" : "/")))

```



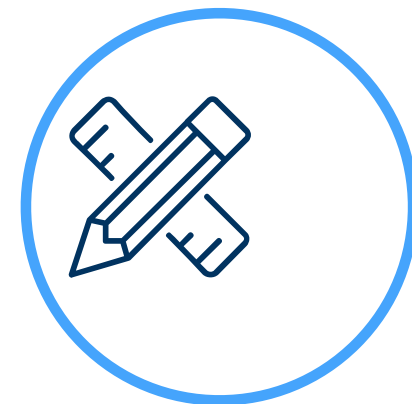
CrownPeak

CrownPeak Extensions

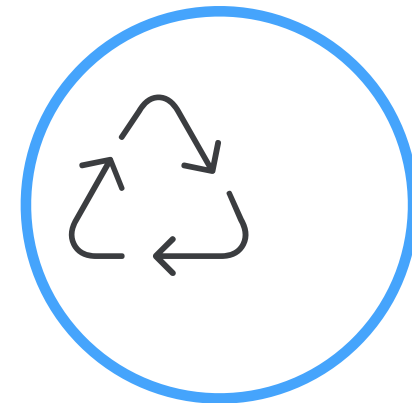
Fahd Shaaban, Director of Professional Services

CrownPeak Extensions Agenda Overview

Welcome to the CrownPeak Extensions Agenda Overview



Introduction to CMS Extensions: 8 minutes



6 Extension Demos: 7 minutes each

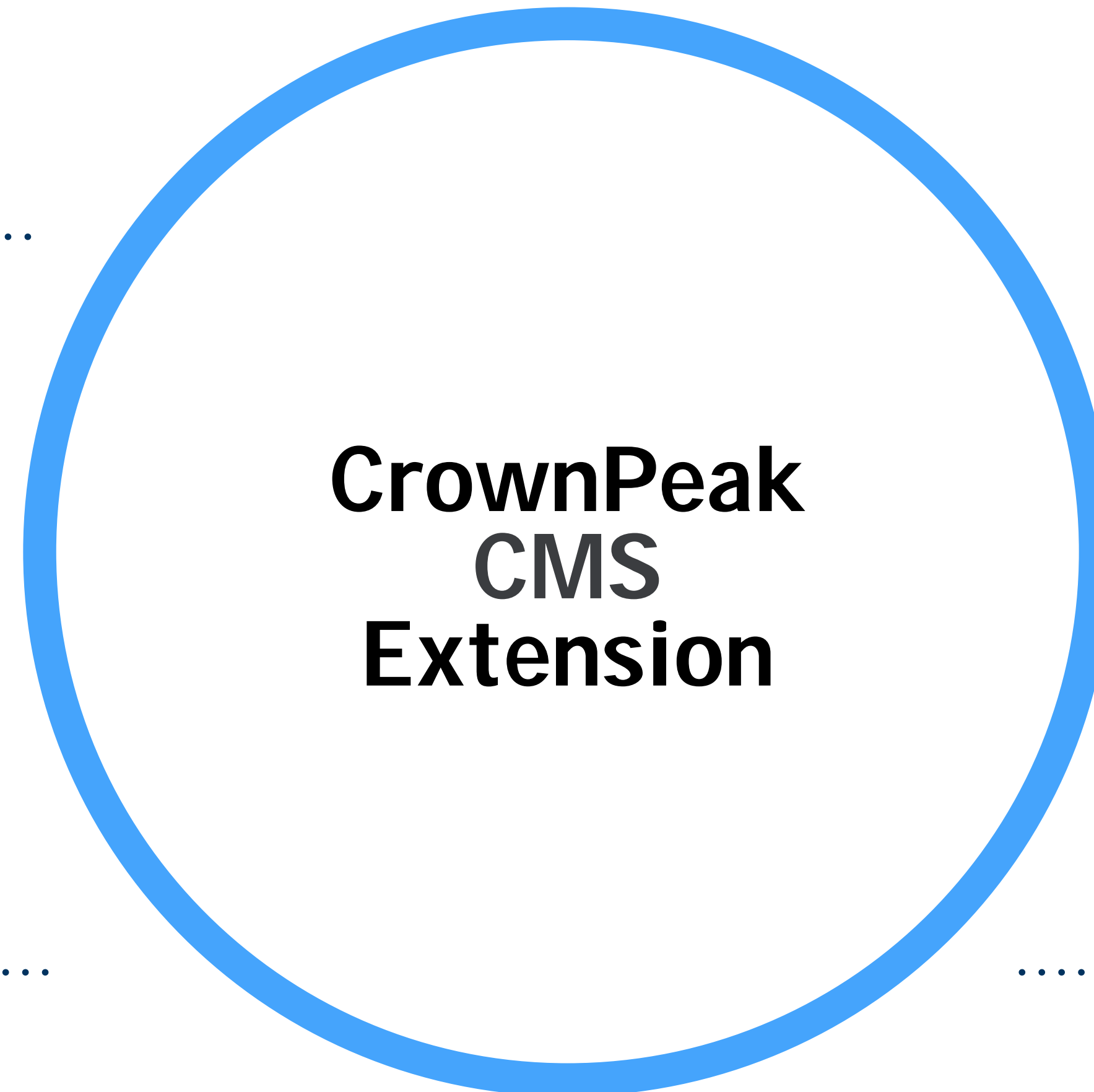


Questions and Your Feedback: 10 minutes



**Introduction
to
CMS Extensions**

What is a CrownPeak Extension?



Templates

Models

Configuration Assets

Library Methods

Workflow Configurations

..... Extends the CMS

..... Best Practices

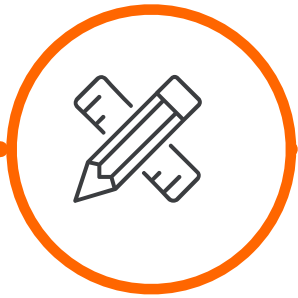
..... Complete Solution

..... Easy to install

..... Easy to configure

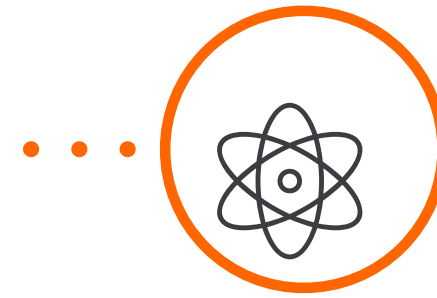
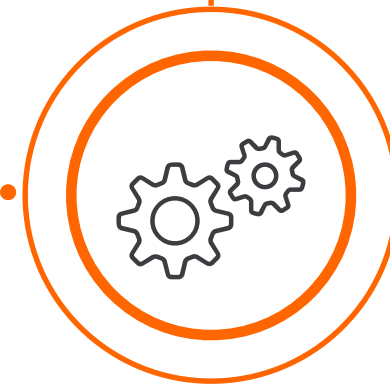
The Path to a CrownPeak Extension and beyond...

Ideas,
Business Problems,
Technical Challenges



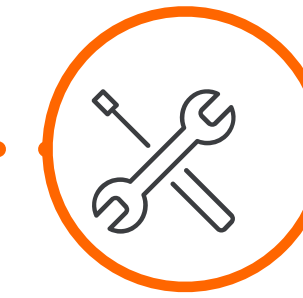
Design and Architect
Solution

Configuration,
Documentation



Package
Configuration
as Extension

Install Extension
and Improve

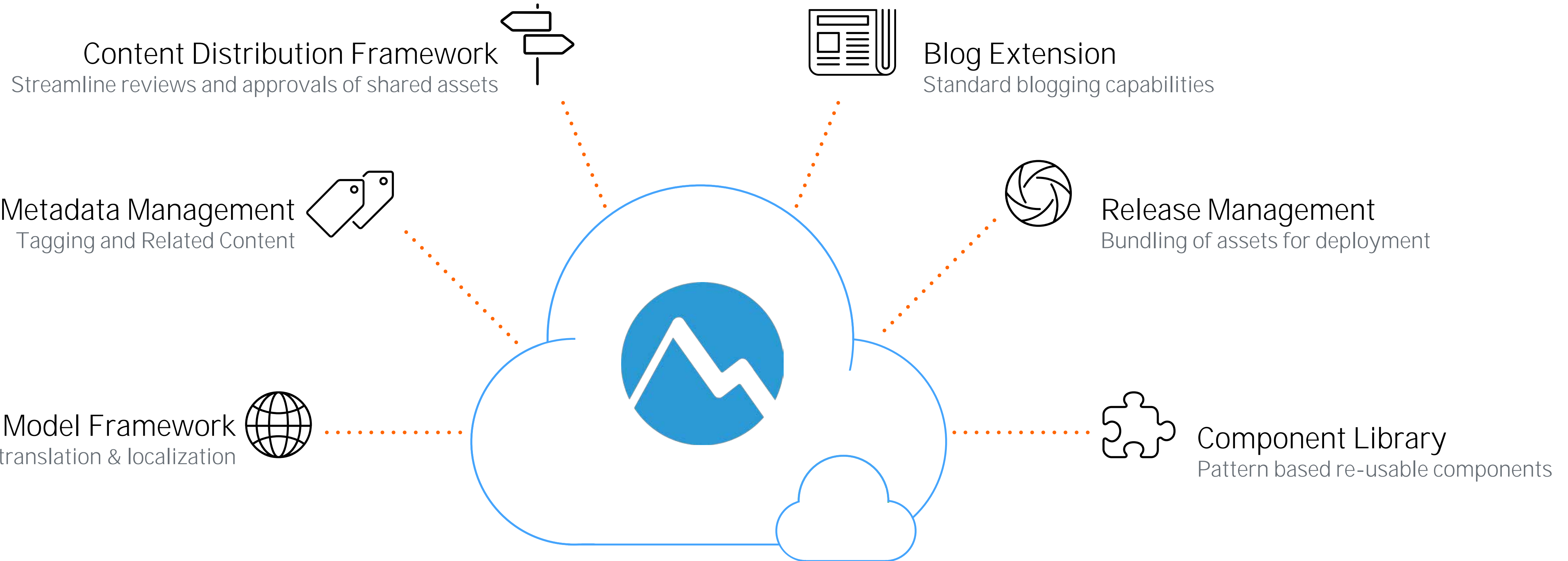


Promote
Library Methods
to Services Library



Current Extensions

More are under development



Release Management

Release Management extends the native workflow and publishing capabilities of the CMS to provide the following

1. Ability to bundle assets for batch deployment based on various criteria
2. Ability to release the bundled assets upon review and approval
3. Audit reports of bundled assets and deployment details

Blog Extension

The Blog extension provides a framework to manage blog posts with the following capabilities

1. Blog posts by contributors
2. Organized by Categories and archived by year/month
3. Support for Tagging
4. Ability to featured posts and popular posts
5. Content subscriptions
6. User comments

Translation Model Framework

The Translation Model Framework, or TMF, provides

1. A framework and a toolset to facilitate content translation and localization in the CMS
2. Ability to derive content from master or source content
3. Ability to notify content owners when content is changed

Content Distribution Framework

The Content Distribution Framework provides

1. A framework and a toolset to facilitate content sharing and content re-use, with the goal of streamlining content authoring, review and approvals
2. Ability to define a common folder of assets as a Universal or Global source for shared content
3. Ability for authors and editors to distribute and localize the shared content into destination sites, regions, etc.
4. Auto-updates of distributed content whenever the original universal/global content is modified

Metadata Management

Metadata Management, or MDM, extends the native capabilities of the CMS to provide the following

1. Multiple Authoring options to capture tags
2. Library Methods to quickly and easily find related assets via applied tags
3. Ability to export tags and related asset references as XML or JSON for CrownPeak Search or 3rd

Party applications for dynamic web experiences

Component Library

The Component Library is a CrownPeak Extension and Approach that provides

1. A process to define re-usable template components containing approved and repeatable front-end “patterns” (HTML/CSS/JS)
2. The corresponding CMS logic necessary to drive
 - a. The authoring experience, which includes content capture, and content validation
 - b. Content Rendering

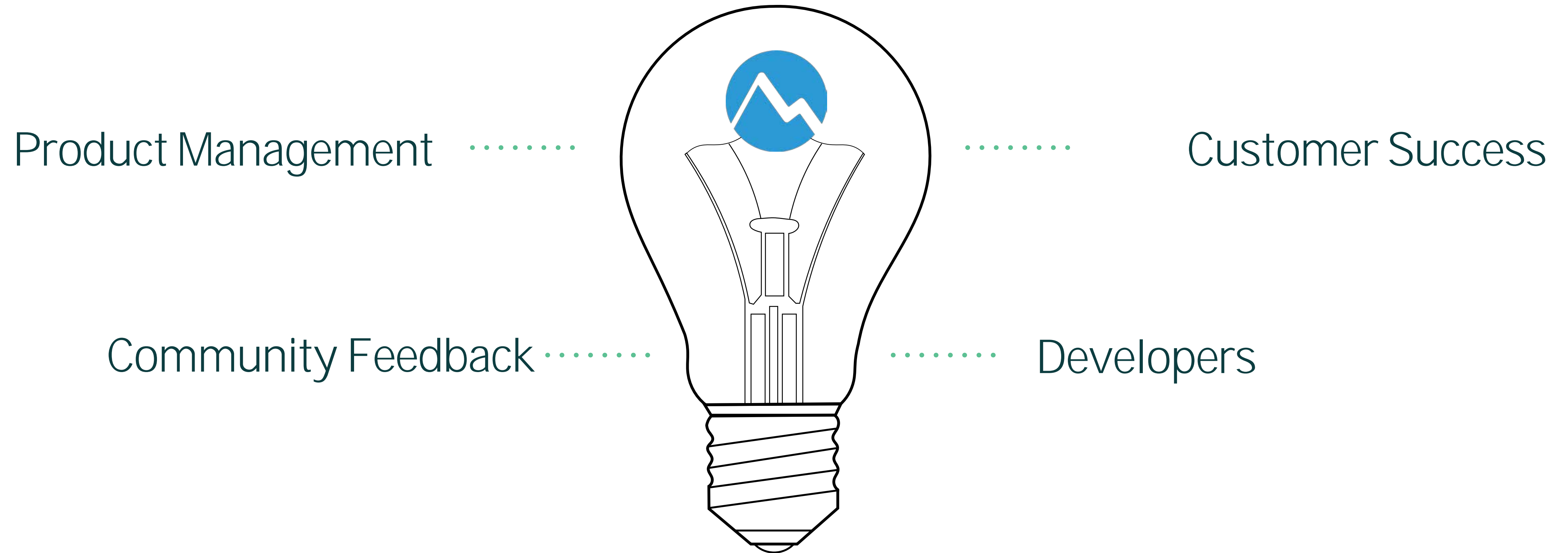


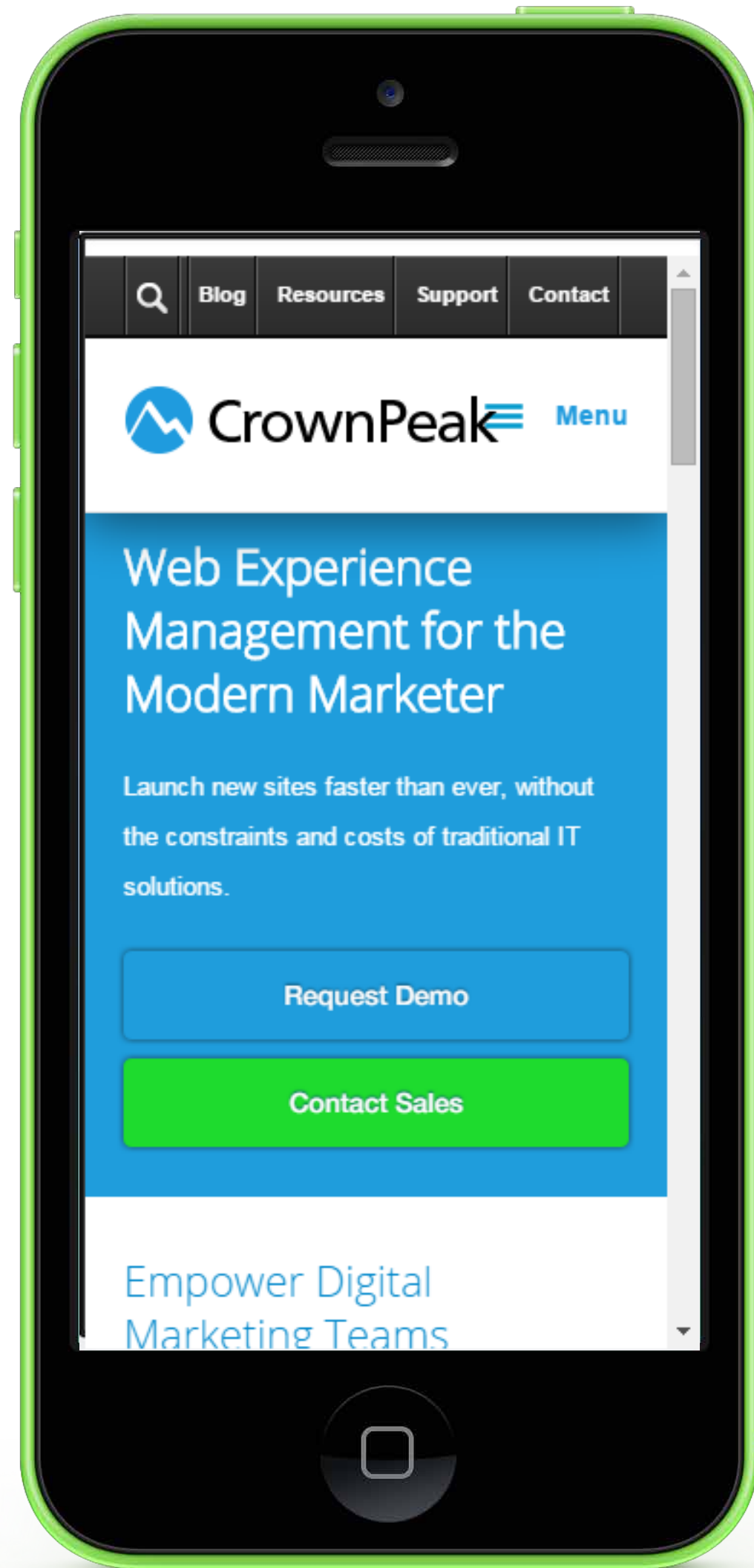
Before we go...

Your Feedback

We want your feedback!

We are here to hear your ideas and requests.





Twitter

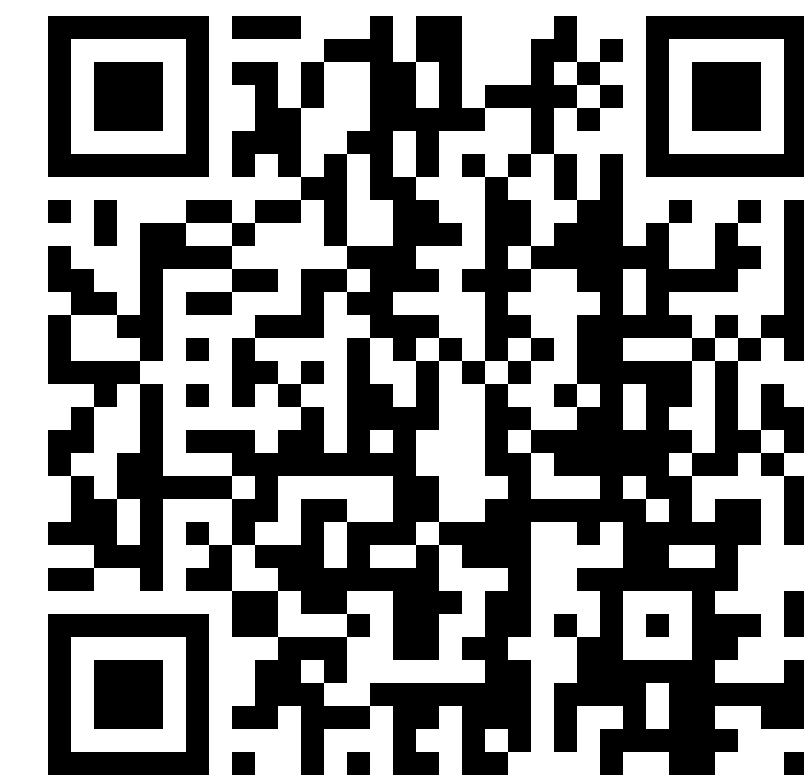
@CrownPeak
#CrownPeakPartners



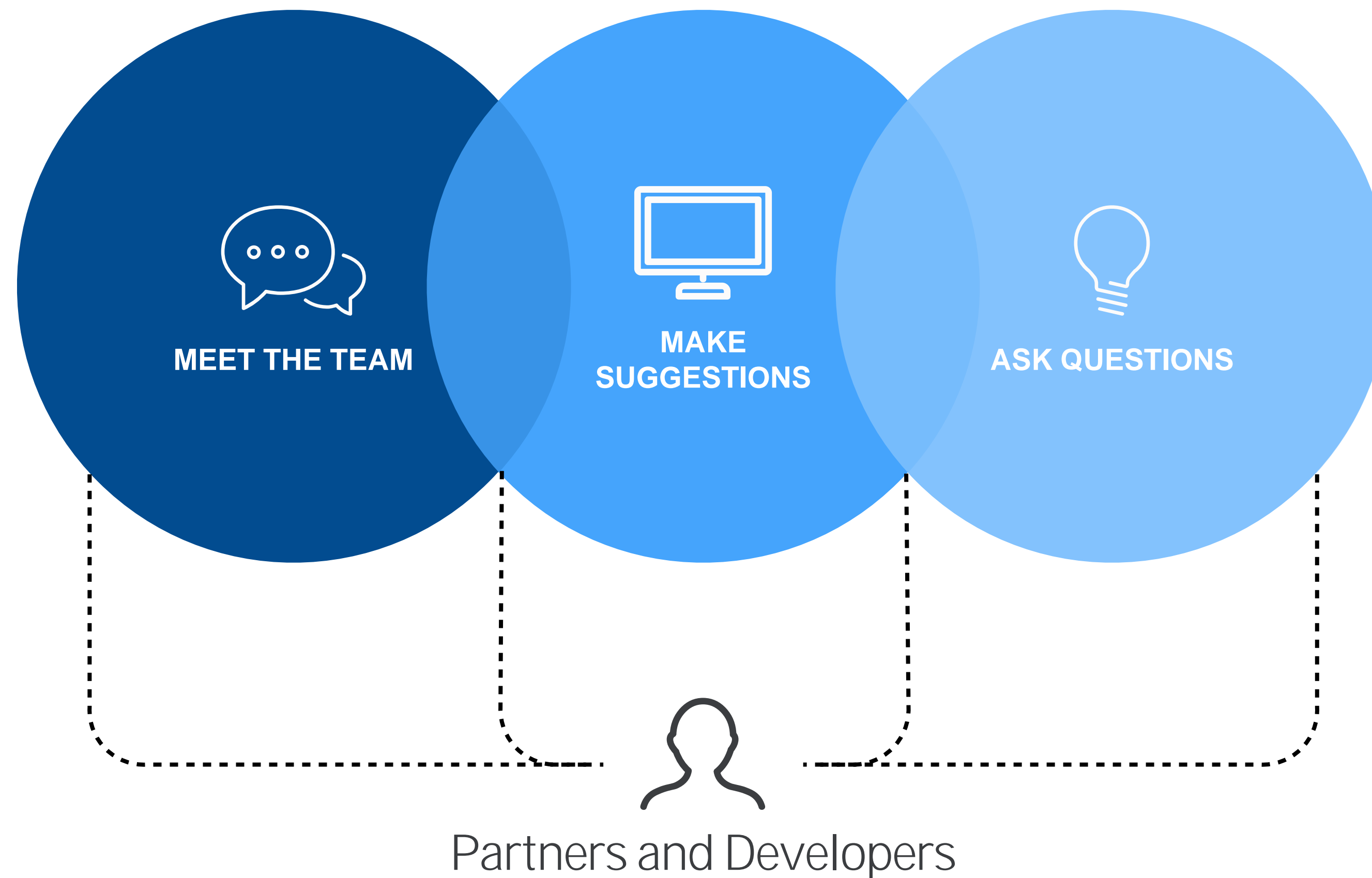
Forums on Connect

For ongoing discussions with the CrownPeak community on

- [Template development](#)
- [User & Developer Experience](#)
- [Solutions and Extensions](#)
- [Product features & roadmap](#)
- [Hosting & Publishing](#)
- [Integrations](#)



Access to Customer Success and Product Teams





Partners and Developers