

# Содержание



# Интеграция ADVANTA с Analytic Workspace

[Analytic Workspace](#) — российская BI-система, позволяющая обрабатывать данные из разных источников и визуализировать их при помощи широкого набора встроенных инструментов. Благодаря интеграции возможно использование инструментария Analytic Workspace для построения дашбордов на основе данных из ADVANTA. Это дает дополнительные возможности для анализа данных и последующего формирования управленческих решений.

Ниже приведена подробная пошаговая инструкция по организации процесса обмена данными между двумя системами. Для реализации обмена у вас должен быть доступ к по крайней мере к демо-версиям данных продуктов.

## 1) Сформировать в ADVANTA LINQ-запросы

Интеграция с Analytic Workspace организована с использованием [LINQ-запросов](#) и [Web-API](#) ADVANTA. Таким образом, для того, чтобы загрузить данные в BI-систему, необходимо предварительно сформировать LINQ-запросы внутри ADVANTA для получения тех данных, которые в дальнейшем будут визуализированы.

Пример LINQ-запроса данных по проектам

```
/* 1. перечень всех проектов (2 типа объектов - ИТ-проект и Организационный проект)
с UID и всеми системными реквизитами (сроки начала/завершения - план, факт,
ответственные (ФИО), статус (текстом), последний базовый план)
- объект-родитель (название)
и пользовательскими реквизитами:
- жизненный цикл проекта
- описание
- масштаб проекта (текстовое значение)
- эффект
- последняя запись из справочника "Отчет о статусе" - все поля
- активная запись из справочника "Бюджет проекта" - только поле "Бюджет (план)"
*/
```

```
var projects = dataContext.Projects
    .Where(p => p is IT_proekt_d46b6d || p is
Organizacionnij_proekt_a3e2fb)
    .OrderBy(p => p.CreationDate)
    .Select(p => new { p.Id,
                    p.Name, ParentName = p.Parent.Name,
                    LifeCycleName =
p.Fields.Zhiznennij_cikl_proekta_b07f57.Name,
                    p.SystemStartDate, p.SystemEndDate,
p.ActualStartDate, p.ActualEndDate,
                    p.BaselinePlanStartDate, p.BaselinePlanEndDate,
                    BossFIO = String.Concat( p.Owner.LastName, " ",
p.Owner.FirstName), RPFIO = String.Concat( p.Responsible.LastName, " ",
p.Responsible.FirstName),
                    Description = p.Fields.Opisanie_caaaaa,
                    ProjectScaleName =
```

```

p.Fields.Masshtab_proekta_04412f.Name,
        ProjectGainName = p.Fields.Effekt_b6c145.Name,
        StatusReport_Date =
p.GetChildren<Otchet_o_stavke_440616>().OrderByDescending(o =>
o.Date).FirstOrDefault().Date,
        StatusReport_Chto_sdelano_problemi_riski =
p.GetChildren<Otchet_o_stavke_440616>().OrderByDescending(o =>
o.Date).FirstOrDefault().Chto_sdelano_problemi_riski_4a11b8,
        StatusReport_Prichina_problemi =
p.GetChildren<Otchet_o_stavke_440616>().OrderByDescending(o =>
o.Date).FirstOrDefault().Prichina_problemi_a3e564.Name,
        StatusReport_StatusName =
p.GetChildren<Otchet_o_stavke_440616>().OrderByDescending(o =>
o.Date).FirstOrDefault().Status_e05ba3.Name,
        BudjetSum =
p.GetChildren<Byudzhet_plan_e5bee7>().Where(b =>
b.Aktivnostj_versii_281a38.Id ==
Classifier_Aktivnostj_versii_625508.Aktivnaya_4f5ba775_Id).Sum(b =>
b.Summa_06ff93)
        });
return projects;

```

Пример LINQ-запроса данных по контрольным точкам

```

/*
2. перечень всех контрольных точек (1 тип объекта - КТО)
с UID и всеми системными реквизитами (как и у проекта),
- UID - проекта-родителя (непрямой родитель)
и пользовательские поля:
- описание (текст)
- результат (текст)
- последняя запись из справочника "Отчет о мероприятии" - все поля
*/
var projects = dataContext.KT0_1ff431_List
    .Select(p => new {
        p.Id,
        p.Name,
        ProjectId =
(Guid?)p.GetParentHierarchy<Project>(false).Where(p => p is IT_proekt_d46b6d
|| p is Organizacionnij_proekt_a3e2fb).FirstOrDefault().Id,
        // ProjectName =
p.GetParentHierarchy<Project>(false).Where(p => p is IT_proekt_d46b6d || p
is Organizacionnij_proekt_a3e2fb).FirstOrDefault().Name,
        p.SystemStartDate, p.SystemEndDate, p.ActualStartDate,
p.ActualEndDate,
        p.BaselinePlanStartDate, p.BaselinePlanEndDate,
BossFIO = String.Concat( p.Owner.LastName, " ",
p.Owner.FirstName), RPFIO = String.Concat( p.Responsible.LastName, " ",
p.Responsible.FirstName),
        Description = p.Opisanie_caajaa,

```

```

Result = p.Rezultat_8e716f,
StatusReport_Date =
p.GetChildren<Otchet_o_statuse_meropriyatiya_2bc942>().OrderByDescending(o
=> o.Date).FirstOrDefault().Date,
StatusReport_Chto_sdelano_problemi_riski =
p.GetChildren<Otchet_o_statuse_meropriyatiya_2bc942>().OrderByDescending(o
=> o.Date).FirstOrDefault().Chto_sdelano_problemi_riski_4a11b8,
StatusReport_Prichina_problemi =
p.GetChildren<Otchet_o_statuse_meropriyatiya_2bc942>().OrderByDescending(o
=> o.Date).FirstOrDefault().Prichina_problemi_a3e564.Name,
StatusReport_StatusName =
p.GetChildren<Otchet_o_statuse_meropriyatiya_2bc942>().OrderByDescending(o
=> o.Date).FirstOrDefault().Status_e05ba3.Name
});
return projects;

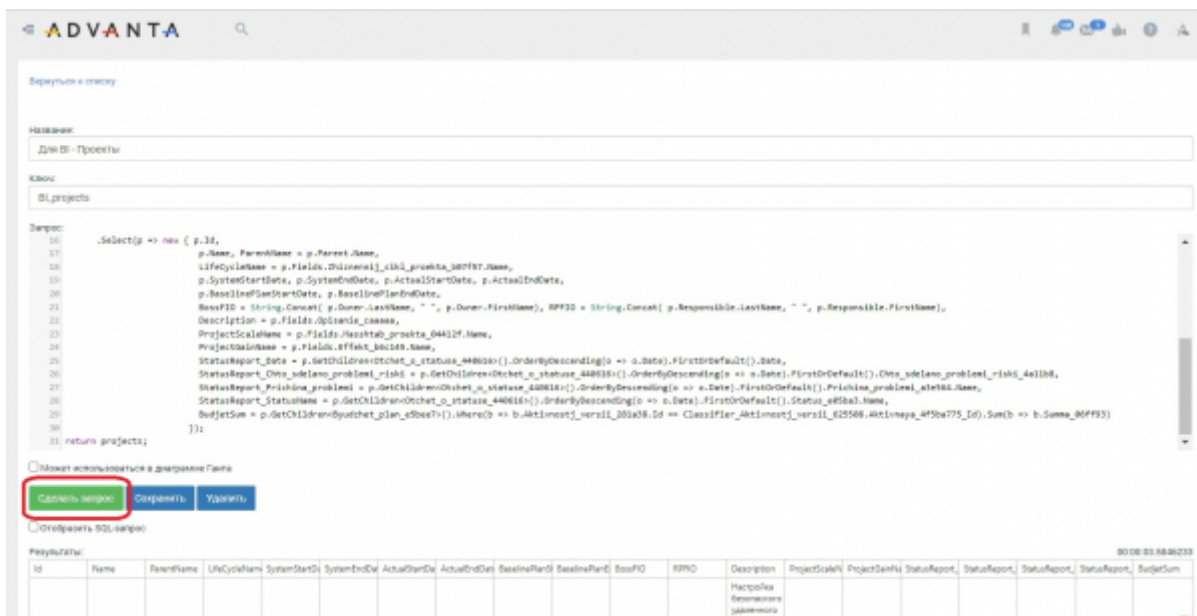
```

Данный код запросов приведен в качестве примера. На вашей инсталляции необходимо будет указать наименования сущностей, объектов, справочников, реквизитов, соответствующих сформированному контексту LINQ-запросов.

## 2) Создать шаблоны таблиц данных в формате .xlsx

Данные шаблоны необходимы для формирования структуры хранения данных в AW в соответствии со структурой получаемых их ADVANTA данных.

Для создания шаблона необходимо выполнить созданный LINQ-запрос в ADVANTA, скопировать шапку полученной таблицы в Excel и сохранить файл в формате .xlsx.



Last update:  
21.04.2024 22:04 product:api:integration\_examples:aw [https://wiki.a2nta.ru/doku.php/product/api/integration\\_examples/aw?rev=1713737067](https://wiki.a2nta.ru/doku.php/product/api/integration_examples/aw?rev=1713737067)

From:  
<https://wiki.a2nta.ru/> - **Wiki [3.x]**

Permanent link:  
[https://wiki.a2nta.ru/doku.php/product/api/integration\\_examples/aw?rev=1713737067](https://wiki.a2nta.ru/doku.php/product/api/integration_examples/aw?rev=1713737067)

Last update: **21.04.2024 22:04**

