



Расширение базовой платформы

Модуль расширяет базовый функционал платформы ELMA, включающее дополнительные библиотеки, функции и методы:

- Режим тестирования/обучения (вход под любым пользователем при наличии соответствующих прав)
- Собственный логотип и название системы на странице авторизации
- Баннер работы в тестовом режиме
- Оповещение пользователей об профилактических работах
- Автосохранение проектных задач
- Перезапуск сервера по расписанию (устранение утечек памяти)
- Много разных хелперов
- Счетчики для объектов системы (аналог нумераторов в документах)
- Входящая почта для пользователей
- Кворум сервисы
- Возможность редактирования комментариев автором

- Настройки

- Автоматическое удаление логов системы

- Хелперы

- ModelHelper

- ListenerHelper
- ServerHelper
- UserHelper
- WorkflowHelper

- Модели
 - LongActionModel

- Пример HTML брэнд шаблона
- Примеры
 - Реализация обработки длительной операции в задаче
 - Скрытие/отображение кнопок переходов в процессных задачах

- Кворум сервисы
 - Мониторинг

- Менеджеры
 - CounterManager

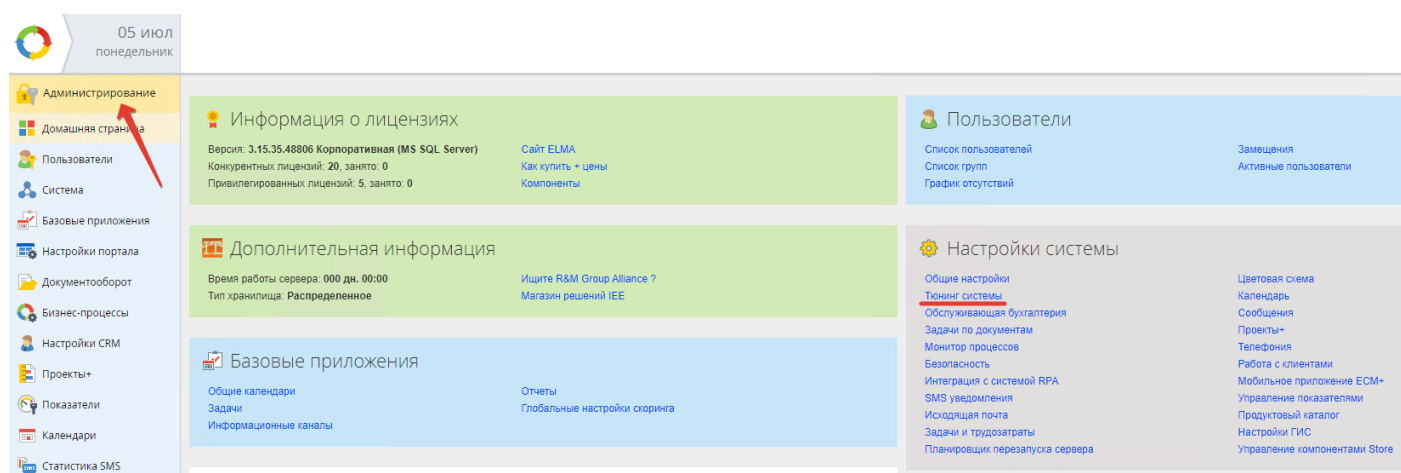
Настройки

Администрирование компонента и тонкие настройки системы

Автоматическое удаление логов системы

Для автоматического удаления старых логов системы необходимо включить данную опцию в настройках:

1. Выбрать меню **Администрирование - Тюнинг системы**.



2. Установить X дней, старше которых логи системы будут удалены, либо 0 для отключения автоматической очистки.

Тюнинг системы

Альтернативное название системы

DEV 3.15

Режим тестирования и обучения

☒ Да
☐ Нет

Время перехода в режим обслуживания

▼

Длительность режима обслуживания (мин.)

15

Начало планового обновления системы

▼

Длительность планового обновления системы

0

▼

ч

1

▼

мин

Дополнительная информация по обновлению

Автоматическая загрузка обновлений

☐ Да
☒ Нет

Получатель системных уведомлений

Администратор ELMA (Адм ✕ + ▼)

Период автосохранения процессных задач (сек.)

30

Включить новогоднее оформление системы

☐ Да
☒ Нет

Обновлять производственный календарь

☒ Да
☐ Нет

Автоматически обновлять глобальный производственный календарь из открытых источников

Хранить логи (дней)

3

Если 0 - хранить всегда. Иначе удалять автоматически старше указанного количества дней

Сохранить

Отмена

Если было установлено количество дней больше 0, то в планировщике появится задание **Удаление старых логов системы**, которое ежедневно будет проверять старые логи и при необходимости удалять их:

05 июл
понедельник

История запусков

Администрирование	Запланированные задания
Домашняя страница	Тип / Триггер / Задание
Пользователи	Статус предыдущего запуска
Система	IEE (ITino ELMA Extensions)
Настройки системы	Триггер запуска (раз в день)
Компоненты	Перечень банков РФ (источник ЦБ РФ)
Внешние приложения	Успешное выполнение
Планировщик	Проверка сроков подписки на компоненты IEE
	Успешное выполнение
	Производственный календарь РФ
	Успешное выполнение
	Удаление старых логов системы
	Успешное выполнение
	Триггер запуска ежемесячных платежей

В результате успешной очистки будет сформирован лог обработки:



05 июл
понедельник



Назад



Администрирование



Домашняя страница



Пользователи



Система



Настройки системы



Компоненты



Внешние приложения



Планировщик



Диагностика системы



Индексирование



Фильтрация объектов



Монитор процессов



Базовые приложения



Настройки портала



Документооборот



Бизнес-процессы



Настройки CRM



Проекты+



Показатели



Календари



Статистика SMS



Логистика



Финансы и бухгалтерия



Настройки СЕМ

Удаление старых логов системы - 05.07.2021 13:40:14

Имя работы Удаление старых логов системы

Имя триггера Триггер запуска (раз в день)

Дата запуска 05.07.2021 13:40

Статус Успешное выполнение

Реальное время запуска 05.07.2021 13:40

Дата завершения 05.07.2021 13:40

Описание Удаление логов старше (дней): 2
C:\ELMA3-Enterprise\WebVlogs\Auth\auth-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\AutoDeployLog\deploy-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\Backup\backup-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\BulkDataImport\BulkDataImport-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\dbUpdate\dbUpdate-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\dbUpdate-Warn\dbUpdate-warn-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\Diagnostics\ElmaRestart\ElmaDiagnostics-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\Feed\feed-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\files\files-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\IEE\accounting-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\IEE\activation-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\IEE\devops-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\IEE\diadoc-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\IEE\gis-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\IEE\sms-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\IEE\social-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\IEE\store-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\IEE\support-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\IEE\telegram-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\IEE\webconf-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\ImportLog\import-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\IndexLog\index-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\Integration1C\1c-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\IntegrationLdapLog\integrationLdap-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\IntegrationOuterData\outerData-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\KPI\kpi-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\MessagesDelivery\msg-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\messaging\messages-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\messaging\sms-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\ModuleActivation\ModuleActivation-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\OrganizationItem\OrganizationItem-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\ProjectMigration\mig-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\Projects\projects-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\Push\push-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\QueueMessage\QM-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\ScriptLog\script-log-20210703 - удален
C:\ELMA3-Enterprise\WebVlogs\script\script-log-20210703 - удален

Хелперы

namespace ITino.ELMA.Common.Helpers

ModelHelper

```
/// <summary>
/// Прописано ли свойство в конфигурации
/// </summary>
/// <param name="pm">Метаданные свойства</param>
/// <returns></returns>
public static bool IsConfigProperty(PropertyMetadata pm)
```

```
/// <summary>
/// Получить Uid значения перечисления
/// </summary>
/// <param name="value">Значение перечисления</param>
/// <returns></returns>
public static Guid GetEnumUid(object value)
```

```
/// <summary>
/// Получить отображаемое описание значения перечисления
/// </summary>
/// <param name="value">Значение перечисления</param>
/// <returns></returns>
public static string GetEnumDescription(object value)
```

```
/// <summary>
/// Название типа сущности
/// </summary>
/// <param name="entity">Сущность</param>
/// <returns></returns>
public static string GetEntityDisplayName IEntity entity)
```

```
/// <summary>
/// Описание типа сущности
/// </summary>
/// <param name="entity">Сущность</param>
/// <returns></returns>
```



```
public static string GetEntityDescription IEntity entity)
```

```
/// <summary>
/// Проверить наличие элемента справочника и создать при необходимости
/// </summary>
/// <param name="uid">Uid элемента справочника</param>
/// <param name="values">Значения свойств</param>
/// <typeparam name="T">Тип сущности</typeparam>
public static void CheckEntityPresent<T>(Guid uid, object values) where T : IEntity
```

```
/// <summary>
/// Удалить элемент справочника (если есть)
/// </summary>
/// <param name="uid">Uid элемента справочника</param>
/// <typeparam name="T">Тип сущности</typeparam>
public static void RemoveEntityIfPresent<T>(Guid uid) where T : IEntity
```

ELMA4

```
/// <summary>
/// Обрезать часть даты и времени
/// </summary>
/// <param name="dateTime"></param>
/// <param name="timeSpan"></param>
/// <returns></returns>
public static DateTime Truncate(this DateTime dateTime, TimeSpan timeSpan)
```

```
/// <summary>
/// Обрезать миллисекунды
/// </summary>
/// <param name="dateTime"></param>
/// <returns></returns>
public static DateTime TruncateMs(this DateTime dateTime)
```

ListenerHelper

```
/// <summary>Получить старое значение</summary>
/// <typeparam name="T">Тип</typeparam>
/// <param name="event">Событие</param>
/// <param name="propertyName">Название свойства</param>
/// <returns>Старое значение</returns>
public static T GetOldValue<T>(PostUpdateEvent @event, string propertyName)
```

```
/// <summary>Получить старое значение</summary>
/// <typeparam name="T">Тип</typeparam>
/// <param name="event">Событие</param>
/// <param name="propertyName">Название свойства</param>
/// <returns>Старое значение</returns>
public static T GetOldValue<T>(PreUpdateEvent @event, string propertyName)
```

```
/// <summary>Получить значение свойства</summary>
/// <typeparam name="T">Тип</typeparam>
/// <param name="event">Событие</param>
/// <param name="propertyName">Название свойства</param>
/// <returns>Значение</returns>
public static T GetValue<T>(PostUpdateEvent @event, string propertyName)
```

```
/// <summary>Присвоить значение свойству</summary>
/// <param name="event">Событие</param>
/// <param name="propertyName">Название свойства</param>
/// <param name="value">Значение</param>
public static void SetValue(PreUpdateEvent @event, string propertyName, object value)
```

```
/// <summary>Получить значение свойства</summary>
/// <typeparam name="T">Тип</typeparam>
/// <param name="event">Событие</param>
/// <param name="propertyName">Название свойства</param>
/// <returns>Значение</returns>
public static T GetValue<T>(PostInsertEvent @event, string propertyName)
```

```
/// <summary>Присвоить значение свойству</summary>
/// <param name="event">Событие</param>
/// <param name="propertyName">Название свойства</param>
/// <param name="value">Значение</param>
public static void SetValue(PreInsertEvent @event, string propertyName, object value)
```

ServerHelper

```
/// <summary>  
/// Получить среднее время запуска сервера  
/// </summary>  
public static string AverageStartTime
```

```
/// <summary>  
/// Версия компонентов IEE  
/// </summary>  
public static string IEEVersion
```

```
/// <summary>  
/// Возвращает признак что система в режиме тестирования  
/// </summary>  
public static bool IsTesting
```

```
/// <summary>  
/// Перезапустить сервер  
/// </summary>  
public static void RestartServer()
```

```
/// <summary>  
/// Логгер импорта  
/// </summary>  
public static ILog ImportLogger
```

```
/// <summary>  
/// Запись в протокол импорта данных в справочник  
/// </summary>  
/// <param name="name">Имя справочника</param>  
/// <param name="total">Всего записей</param>  
/// <param name="new">Новых записей</param>  
/// <param name="update">Обновлений</param>  
/// <param name="delete">Удалений</param>
```

```
public static void ImportLogCatalog(string name, int total, int @new = 0, int update = 0, int delete = 0)
```

```
/// <summary>  
/// Запись в протокол импорта что данные для справочника отсутствуют  
/// </summary>  
/// <param name="name">Имя справочника</param>  
public static void ImportLogCatalogNone(string name)
```

```
/// <summary>  
/// Время работы сервера  
/// </summary>  
/// <returns></returns>  
public static TimeSpan UpTime
```

```
/// <summary>  
/// Логгер обновления БД  
/// </summary>  
public static ILog DbUpdateLogger
```

```
/// <summary>  
/// Логер пакетной обработки  
/// </summary>  
public static ILog ProcessLogger
```

UserHelper

```
/// <summary>
/// Системный пользователь
/// </summary>
public static IUser SystemUser
```

```
/// <summary>
/// Пользователь подсистемы обмена
/// </summary>
public static IUser ExchangeUser
```

```
/// <summary>
/// Получить непосредственного руководителя для пользователя
/// </summary>
/// <param name="user">Пользователь</param>
/// <returns>Руководитель (если есть, иначе null)</returns>
public static IUser GetImmediateChiefByUser(IUser user)
```

```
/// <summary>
/// Получить всех активных пользователей группы
/// </summary>
/// <param name="gUidStr">Строковый Uid группы</param>
/// <returns>Список пользователей</returns>
public static IEnumerable<IUser> GetUsersInGroup(string gUidStr)
```

```
/// <summary>
/// Получить всех активных пользователей группы
/// </summary>
/// <param name="gUid">Uid группы</param>
/// <returns>Список пользователей</returns>
public static IEnumerable<IUser> GetUsersInGroup(Guid gUid)
```

```
/// <summary>
/// Преобразовать полные инициалы в сокращенные
```

```
/// </summary>
/// <param name="fullName">Полные инициалы</param>
/// <param name="lastNameOrder">Фамилия в начале</param>
/// <returns>Сокращенные инициалы</returns>
public static string ConvertToShortName(string fullName, bool lastNameOrder = true)
```

```
/// <summary>
/// Входит ли пользователь в группу Администраторы
/// </summary>
/// <returns></returns>
public static bool IsAdmin
```

Только ELMA4

```
/// <summary>
/// Отправить push уведомление текущему пользователю
/// </summary>
/// <param name="title">Заголовок уведомления</param>
/// <param name="message">Сообщение</param>
/// <param name="clickUrl">Ссылка в браузере при нажатии</param>
/// <param name="tag">Тэг (для группировки)</param>
/// <returns></returns>
public static bool WebPush(string message, string title = null, string clickUrl = null, string tag = null)
```

```
/// <summary>
/// Отправить push уведомление пользователю
/// </summary>
/// <param name="user">Пользователь</param>
/// <param name="title">Заголовок уведомления</param>
/// <param name="message">Сообщение</param>
/// <param name="clickUrl">Ссылка в браузере при нажатии</param>
/// <param name="tag">Тэг (для группировки)</param>
/// <returns></returns>
public static bool WebPush(IUser user, string message, string title = null, string clickUrl = null, string tag = null)
```

WorkflowHelper

```
/// <summary>
/// Получить запускаемые процессы по входному типу сущности
/// </summary>
/// <param name="typeUid">UID типа сущности</param>
/// <returns>Список процессов Workflow</returns>
public static IEnumerable<IWorkflowProcess> GetStartableProcessesByInputEntity(Guid typeUid)
```

```
/// <summary>
/// Комментарий для процессной задачи
/// </summary>
/// <param name="element">Uid элемента задачи на схеме</param>
/// <param name="task">Задача</param>
/// <param name="author">Автор комментария</param>
/// <param name="date">Дата комментария</param>
/// <param name="comment">Комментарий</param>
/// <param name="type">Тип комментария</param>
public static void CommentForTask(Guid element, ITaskBase task, IUser author, DateTime date, string comment,
EOWorkflowTaskCommentType type = EOWorkflowTaskCommentType.Info)
```

```
/// <summary>
/// Сохранить уведомление для последующего показа
/// </summary>
/// <param name="message">Текст сообщения</param>
/// <param name="type">Тип сообщения</param>
public static void Notify(string message, NotifyType type = NotifyType.Info)
```

```
/// <summary>
/// Убрать замещение у задачи процесса
/// </summary>
/// <param name="task">Задача процесса</param>
/// <param name="uid">Uid задачи на схеме</param>
/// <param name="ignoreUsers">Список игнорируемых пользователей</param>
public static void RemoveReplacement([NotNull] this ITaskBase task, Guid uid, IEnumerable<IUser> ignoreUsers
```



```
= null)
```

```
/// <summary>  
/// Завершить процесс с выводом ошибки пользователю  
/// </summary>  
/// <param name="context"></param>  
/// <param name="error"></param>  
public static void CompleteWithError(object context, string error)
```

```
/// <summary>  
/// Выполнить длительное действие в отдельном потоке при создании задачи  
/// </summary>  
/// <param name="element">Id задачи на диаграмме</param>  
/// <param name="task">Созданная задача в OnTaskCreate</param>  
/// <param name="action">Действие</param>  
public static void ProcessLongAction(Guid element, ITaskBase task, Action<LongActionModel, ISession> action)
```

Модели

LongActionModel

```
public class LongActionModel
{
    /// <summary>
    /// Логгер
    /// </summary>
    public ILog Logger { get; }

    /// <summary>
    /// Контекст процесса
    /// </summary>
    public WorkflowInstanceContext Context { get; }

    /// <summary>
    /// Начало обработки
    /// </summary>
    public DateTime StartDate { get; }

    /// <summary>
    /// Окончание обработки
    /// </summary>
    public DateTime? EndDate { get; set; }

    /// <summary>
    /// Название действия
    /// </summary>
    public string Name { get; set; }

    /// <summary>
    /// Задача бизнес процесса
    /// </summary>
    public IWorkflowTaskBase Task { get; set; }

    /// <summary>
```

```
/// Кнопки переходов
```

```
/// </summary>
```

```
public IEnumerable<string> Buttons { get; set; }
```

```
/// <summary>
```

```
/// Ошибка выполнения
```

```
/// </summary>
```

```
public string Error { get; set; }
```

```
/// <summary>
```

```
/// Обновить задачу в браузере после завершения длительной операции
```

```
/// </summary>
```

```
public bool RefreshOnCompleted { get; set; }
```

```
}
```

Пример HTML брэнд шаблона

ELMA4

```
<div style="font-family: Arial, Helvetica, sans-serif; font-weight: 100;">
<table style="border-spacing: 0px; border: 0; width: 100%; border-collapse: collapse;">
<tr style="background: #ffdddd; text-align: right;">
<td style="width: 1%; padding: 10px;">

</td>
<td style="width: 99%; padding: 15px; vertical-align: top;">
<span style="font-size: 2.0em;">Название Вашей системы</span>
</td>
</tr>
<tr>
<td colspan="2" style="background: #6495ed; font-size: 1.2em; padding: 10px;">
<div>{$Body}</div>
</td>
</tr>
<tr>
<td colspan="2" style="color: gray; font-size: 0.7em; padding-top: 10px;">
<div>УВЕДОМЛЕНИЕ О КОНФИДЕНЦИАЛЬНОСТИ: Это электронное сообщение и любые документы,
приложенные к нему, содержат конфиденциальную информацию. Настоящим уведомляем Вас о том, что
если это сообщение не предназначено Вам, использование, копирование, распространение информации,
содержащейся в настоящем сообщении, а также осуществление любых действий на основе этой
информации, строго запрещено. Если Вы получили это сообщение по ошибке, пожалуйста, сообщите об
этом отправителю по электронной почте и удалите это сообщение.</div>
</td>
</tr>
</table>
</div>
```

Результатом будет:

Тестовое письмо



ELMA-DEV8 <no-reply@randm.ru>

9:52



То: Администратор ELMA



Данные.dat
1.37 MB



ВАШЕ
ЛОГО

Название Вашей системы

Это тестовое сообщение по вашему запросу --

Это сообщение автоматически сформировано системой ELMA

УВЕДОМЛЕНИЕ О КОНФИДЕНЦИАЛЬНОСТИ: Это электронное сообщение и любые документы, приложенные к нему, содержат конфиденциальную информацию. Настоящим уведомляем Вас о том, что если это сообщение не предназначено Вам, использование, копирование, распространение информации, содержащейся в настоящем сообщении, а также осуществление любых действий на основе этой информации, строго запрещено. Если Вы получили это сообщение по ошибке, пожалуйста, сообщите об этом отправителю по электронной почте и удалите это сообщение.

Примеры

Полезные примеры использования элементов компонента

Реализация обработки длительной операции в задаче

Для обработки длительных операций в системе, когда использование блока "Скрипт" приводит к таймауту и срыву процесса, можно использовать следующий метод. При этом в самой задаче будет отображаться прогресс обработки, на время обработки кнопки переходов будут скрыты.

Внутри обработчика нельзя использовать **context**. Вместо этого использовать **model.Context**.

```
public override void OnTaskCreate(ITaskBase task, P_IA_AandP context)
{
    // 4f55dd98-5242-4633-88e8-4c745551b52c - Id задачи на диаграмме
    WorkflowHelper.ProcessLongAction(new Guid("4f55dd98-5242-4633-88e8-4c745551b52c"), task, (model,
    session) =>
    {
        model.Name = "Сбор данных по активным задачам"; // Название операции, отображаемое в задаче

        // Здесь любой код для длительных операций
        var instanceContext = model.Context as P_IA_AandP;
    });
}
```

В ELMA4 изменена сигнатура вызова метода - дополнительно передается тип контекста:

```
public override void OnTaskCreate(ITaskBase task, P_IA_AandP context)
{
    // 4f55dd98-5242-4633-88e8-4c745551b52c - Id задачи на диаграмме
    WorkflowHelper.ProcessLongAction<P_IA_AandP>(new Guid("4f55dd98-5242-4633-88e8-4c745551b52c"), task,
    (model, session) =>
```



```
{
```

```
model.Name = "Сбор данных по активным задачам"; // Название операции, отображаемое в задаче
```

```
// Здесь любой код для длительных операций
```

```
var instanceContextVar1 = model.Context.Var1;
```

```
});
```

```
}
```

После окончания длительной операции контекст процесса сохранять не нужно - это будет сделано автоматически, если были изменения.

В результате при создании задачи запускаться в отдельном потоке обработка, кнопки переходов будут скрыты и будет отображена информация об операции со счетчиком времени выполнения. Например:

Анализ по ОИ

Сбор данных по активным задачам: 0.00:00:40

> Информация о процессе

Главная страница

История

Динамические ссылки

Сводная таблица по ОИ

ОИ	Текущий статус
ALK034077	Отмена РБ
ALK034083	Отказ клиента
ALK034087	Отказ клиента
YAR033999	4 Передано в Строительство

После завершения длительно операции кнопки переходов вновь появятся и будет выдано сообщение со временем завершения выполнения указанной операции:

Сделано

Работа начата

Действия

Карта процесса

Улучшить

Анализ по ОИ

Сбор данных по активным задачам (завершено 31.08.2021 18:25:43)

> Информация о процессе

Главная страница

История

Динамические ссылки

Сводная таблица по ОИ

ОИ ^	Текущий статус
ALK034077	Отмена РБ
ALK034083	Отказ клиента
ALK034087	Отказ клиента
YAR033999	4 Передано в Строительство

-> Событие 2

Если в исполняемом коде будет вызвано исключение, то появится соответствующее сообщение, кнопки переходов так же появятся:

Сделано

Работа начата

Действия

Карта процесса

Улучшить

Анализ по ОИ

Пример исключения при обработке

> Информация о процессе

Главная страница

История

Динамические ссылки

Сводная таблица по ОИ

ОИ ^	Текущий статус
ALK034077	Отмена РБ
ALK034083	Отказ клиента
ALK034087	Отказ клиента
YAR033999	4 Передано в Строительство

-> Событие 2

Скрытие/отображение кнопок переходов в процессных задачах

Начиная с 4.0.28 и 3.15.48

На текущий момент ELMA4 P2 не поддерживается

Для скрытия или отображения в сценарии процесса кнопок перехода в задачах можно использовать метод `form.HideConnector`:

```
using ITino.ELMA.Common.Helpers;

/// <summary>
/// Скрыть кнопку перехода на какое то изменение
/// </summary>
/// <param name="context">Контекст процесса</param>
/// <param name="form"></param>
public virtual void HideOnChange (Context context, EleWise.ELMA.Model.Views.FormViewBuilder<Context>
form)
{
    /// Что то меняем в значениях контекста процесса (для примера)
    context.Two = context.One;

    /// Скрыть кнопку перехода, где Uid перехода на диаграмме процесса
    form.HideConnector(new Guid("b29c9cc7-7068-4213-a6e3-987171a702aa"));
}

/// <summary>
/// Показать ранее скрытую кнопку перехода на какое то изменение
/// </summary>
```

```

/// <param name="context">Контекст процесса</param>
/// <param name="form"></param>
public virtual void ShowOnChange (Context context, EleWise.ELMA.Model.Views.FormViewBuilder<Context>
form)
{
    // Показать кнопку перехода, где Uid перехода на диаграмме процесса
    form.HideConnector(new Guid("b29c9cc7-7068-4213-a6e3-987171a702aa"), false);
}

```

Все кнопки доступные изначально:

И для ELMA4:

После выполнения скрипта, если есть хоть один скрытый переход, то в верхнем меню убираем **Сделано**, т.к. невозможно динамически определить в нем нужных переход:

Кворум сервисы

Распределенные отказоустойчивые сервисы в кластере Корпоративной редакции системы

Мониторинг

Доступно только в Корпоративной редакции

Статус установленных кворум сервисов кластера отображается по ссылке

Администрирование - Система - Кворум сервисы:

Администрирование	Кворум сервисы		
Домашняя страница	Для показа актуального состояния сервисов компоненты должны быть установлены на всех узлах кластера!		
Пользователи	Сервис / Узел кластера	Статус сервиса	Версия сервиса
Система	Чат бот Telegram		
Настройки системы	e3ent-3	Старая версия	3.15.42.211023
Компоненты	e3ent-2	Активен	3.15.42.211029
Внешние приложения	e3ent-1	Пассивен	3.15.42.211029
Планировщик			
Диагностика системы			
Индексирование			
Фильтрация объектов			
Монитор процессов			
Кворум сервисы			

Информация о текущих пакетах на обработку кворум сервисами находится по ссылке

Справочники - Элемент очереди кворумного сервиса.

Менеджеры

CounterManager

```
public class CounterManager : EntityManager<ICOCOUNTER, long>
{
    /// <summary>
    /// instance
    /// </summary>
    public new static CounterManager Instance => Locator.GetServiceNotNull<CounterManager>();

    /// <summary>
    /// Загрузить или создать новый
    /// </summary>
    /// <param name="uid">Уникальный идентификатор счетчика</param>
    /// <param name="name">Название</param>
    /// <param name="mask">Маска в формате string.Format</param>
    /// <returns></returns>
    public ICOCOUNTER LoadOrCreate(Guid uid, string name = null, string mask = null);

    /// <summary>
    /// Получить новое значение по маске
    /// </summary>
    /// <param name="counter">Счетчик</param>
    /// <param name="save">Сохранить в базе</param>
    /// <param name="checkYear">Сбросить для нового года</param>
    /// <param name="args">Дополнительные аргшуметы для маски</param>
    /// <returns></returns>
    /// <exception cref="Exception"></exception>
    public string FormatNewValue(ICOCOUNTER counter, bool save = true, bool checkYear = false, params object[]
args);

    /// <summary>
    /// Получить новое значение
    /// </summary>
    /// <param name="counter">Счетчик</param>
    /// <param name="save">Сохранить в базе</param>
```

```
/// <param name="checkYear">Сбросить для нового года</param>
/// <returns></returns>
/// <exception cref="ArgumentException"></exception>
[Transaction]
public virtual long GetNewId(ICOCounter counter, bool save = true, bool checkYear = false);
}
```