

# EDDL- Electronic Device Description Language

**César Cassiolato**  
Gerente de Produtos - Smar Equipamentos Industriais Ltda



## Introdução

Quando falamos em protocolos digitais abertos, um dos tópicos mais importantes para o sucesso do interfaceamento entre os equipamentos de campo e os sistemas são os arquivos que irão traduzir as características dos dispositivos e facilitar a visualização, configuração, operação e manutenção para os usuários.

Em 2004 a EDDL, Electronic Device Description Language, que inicialmente vinha sendo o padrão adotado pelo Profibus, foi determinada pela IEC como o padrão internacional para descrição de dispositivos, a chamada IEC 61408-2. Com isto a EDDL pode ser usada para descrever equipamentos HART, Foundation Fieldbus e Profibus e quem ganhou foi o usuário, com maior liberdade de escolha, onde no mercado existirá à sua disposição uma vasta gama de equipamentos e soluções de diversos fabricantes interoperáveis e intercambiáveis.

Os fornecedores de sistemas e equipamentos também são beneficiados pela concorrência mais justa, uma vez que seus sistemas podem suportar equipamentos da concorrência e por outro lado, seus equipamentos de campo podem ser utilizados em sistemas de diferentes fabricantes, ampliando assim a aplicabilidade e, portanto, o seu mercado de forma geral.

### ***Mas afinal, o que é a EDDL?***

A EDDL é uma linguagem baseada em texto, muito parecida com a linguagem C em termos de estruturação, que descreve as características de comunicação digital dos parâmetros dos equipamentos e dispositivos de campo. É utilizada para facilitar a informação e condições de status, diagnósticos e configuração. Sua base é a DDL (Device Description Language utilizada pela HART desde 1992) onde foram acrescentados comandos visuais, principalmente relativos a parte gráfica e imagens e que visam uma melhor interface aos usuários em termos de configuração, calibração e manutenção. Alguns equipamentos, como por exemplo, os posicionadores possuem várias informações que podem ser gráficas, tais como curvas de tendências, assinaturas de válvulas, etc, que agora poderão ser desenvolvidas com mais facilidade e com mais recursividades.

Além disso, a EDDL permite que os fabricantes de Sistemas possam criar um ambiente único e integrado, suportando qualquer equipamento, de qualquer fornecedor e de diferentes protocolos, sem a necessidade de drivers ou arquivos customizados e, aqui está a grande vantagem para o usuário que poderá trabalhar em um ambiente simples, sem a necessidade de treinamentos específicos para cada tipo de protocolo ou ferramenta.

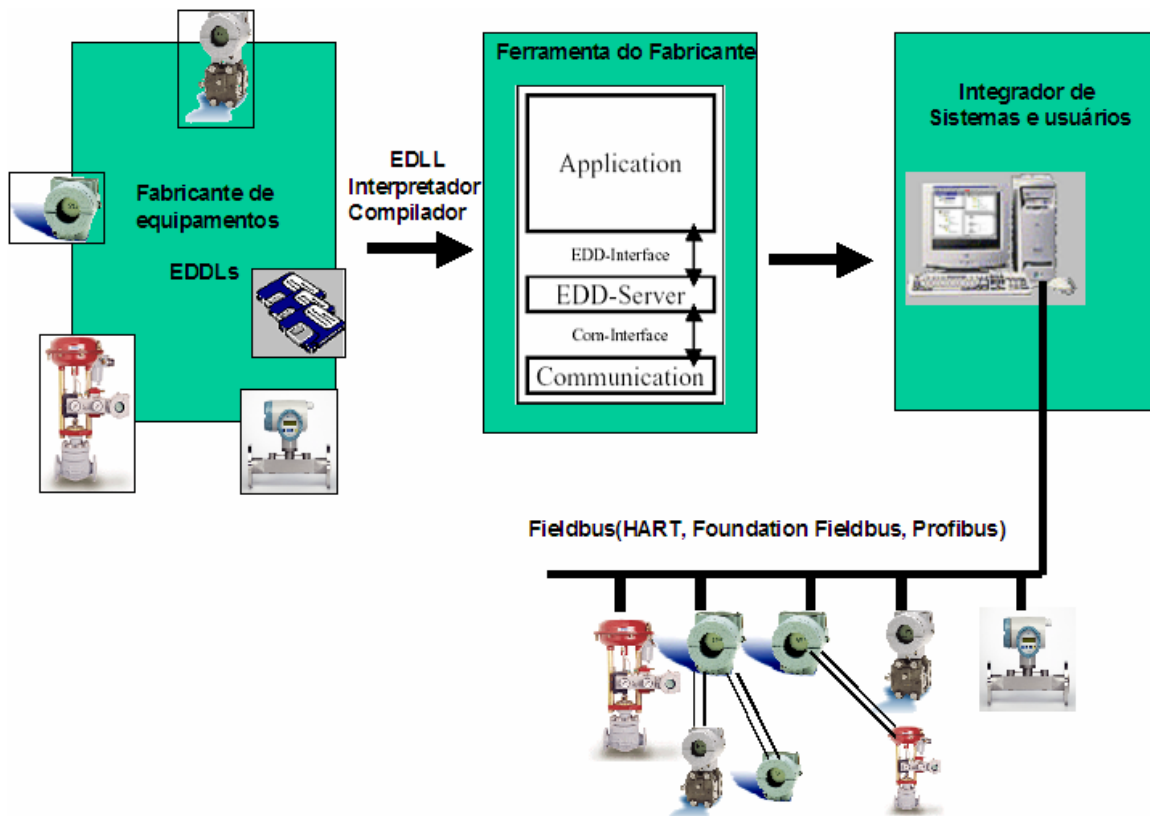


Figura 1 – Integração da EDDL em um sistema Fieldbus

### ***Uma visão geral da EDDL***

A EDDL possui algumas construções básicas onde podemos citar: arrays, blocks, collections, commands, domains, item arrays, menus, methods, programs, records, refresh relations, response codes, unit relations, variable lists, variables, write as one relations.

Vamos a seguir dar alguns exemplos.

#### 1) Variable

```
VARIABLE trd1_self_calibration_cmd
{
    LABEL          [trans_act_self_calibration_cmd];
    HELP           [trans_act_self_calibration_cmd_help];
    CLASS          CONTAINED;
    TYPE           ENUMERATED (1)
    {
        DEFAULT_VALUE 0;
        { 0, "No reaction of the field device" },
        { 2, "Start self calibration / Initialization"},
        { 7, "Reset total valve travel"},
        {255, "Abort current calibration-procedure"}
    }
    HANDLING      READ & WRITE;
}
}
```

Note que na estrutura da variável existem algumas informações que não serão vistas pelo usuário e sim serão interpretadas pela ferramenta de configuração. Toda variável possui os campos de LABEL e HELP que trarão informações sobre o parâmetro, possui uma CLASSE, onde o parâmetro pode ser de uso geral, de entrada, saída, diagnóstico, etc; possui um TYPE, que pode ser

inteiro, float, enumerated, string, etc; e a forma de acesso permitida, escrita ou leitura ou ambos(HANDLING).

## 2) Métodos

```
METHOD method_lower
{
    LABEL [smar_cal_point_lo];
    HELP " Testing.. ";
    DEFINITION
    {
        char    flagtest1;
        int     CalControl;
        float   trim_point, var_feedback;
        int     rc,result;

        flagtest1 = 0;
        trim_point = 0.0;
        /* display a message that requires user acknowledgment */
        rc = ACKNOWLEDGE("WARNING: Control loop should be in manual !");
        if (rc==0)
        {
            fassign(trd1_cal_point_lo,trim_point);
            WriteCommand(write_trd1_cal_point_lo);
        }
        else
            flagtest1 = 1;
        while (flagtest1 == 0)
        {
            ACKNOWLEDGE("Wait the valve stabilize in the position!");
            rc=GET_LOCAL_VAR_VALUE("Please, enter the valve's position:",var_feedback);
            if(rc==0)
            {
                fassign(trd1_feedback_cal,var_feedback);
                WriteCommand(write_trd1_feedback_cal);
                result = SELECT_FROM_LIST("Proceed it again ?", "Yes;No");
                if (result == 0)
                    flagtest1 = 0;
                else
                {
                    CalControl = 0;
                    iassign(trd1_cal_control,CalControl);
                    WriteCommand(write_trd1_cal_control);
                    ACKNOWLEDGE("WARNING: Loop may be returned to last operation
mode !");
                    flagtest1 = 1;
                }
            }
            else
            {
                CalControl = 0;
                iassign(trd1_cal_control,CalControl);
                WriteCommand(write_trd1_cal_control);
                ACKNOWLEDGE("WARNING: Loop may be returned to last operation mode !");
                flagtest1 = 1;
            }
        }
    }
}
```

O exemplo acima mostra um método de calibração de um posicionador de válvula. O Método é a rotina que passo a passo permite a interface entre o equipamento e o usuário.

## 3) Menus

```
MENU Page_trd_set_diag
{
    LABEL "Settings";
    ITEMS
    {
```

```

        Group_setpoint_ao,
        Group_travel_fy303,
        trd1_diagnoses_status,
        method_set,
        BarVal_display_trd,
        BarVal_display_readback
    }
}

```

Os menus são agrupamentos de outros elementos e até mesmo outros menus, normalmente relacionados para uma determinada tarefa específica, onde o usuário pode navegar.

#### 4) Elementos Gráficos

##### a) Bargraph

```

MENU BarVal_display_trd_FY303
{
    LABEL          [trans_act_positioning_value];
    ITEMS
    {
        trd1_positioning_value          (READ_ONLY),
        func3_AO_out_lower_range_value (READ_ONLY),
        func3_AO_out_upper_range_value (READ_ONLY)
    }
}

```

O Bargraph permite que se crie barras de progresso associadas às variáveis, como no exemplo, para facilitar a visualização da posição real da válvula no FY303, posicionador Smar.

##### b) Gráficos

```

MENU Page_yt1
{
    LABEL "Settings and Diagram";
    ITEMS
    {
        Group_setpoint,
        Val_yt,
        OCX_out_display_1,
        Val_temp,
        OCX_out_display_temp
    }
}

```

Os gráficos permitem através de componentes OCX as telas de tendências, os gráficos X/Y, etc..No exemplo, tem-se um gráfico do FY303, onde pode-se acompanhar o valor de setpoint versus a posição real da válvula.

### ***Facilitando a visualização, configuração e manutenção***

A figura 2 mostra uma tela de configuração, bem simples onde o usuário facilmente pode ter acessos a vários parâmetros.

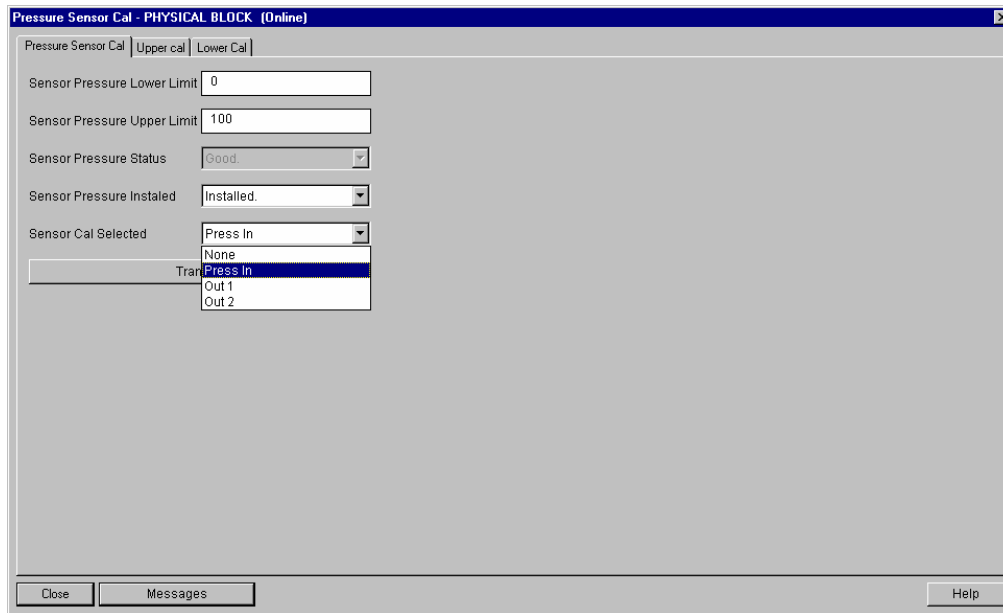


Figura 2 – Variáveis e Menus

A figura 3 dá um exemplo da utilização dos status.

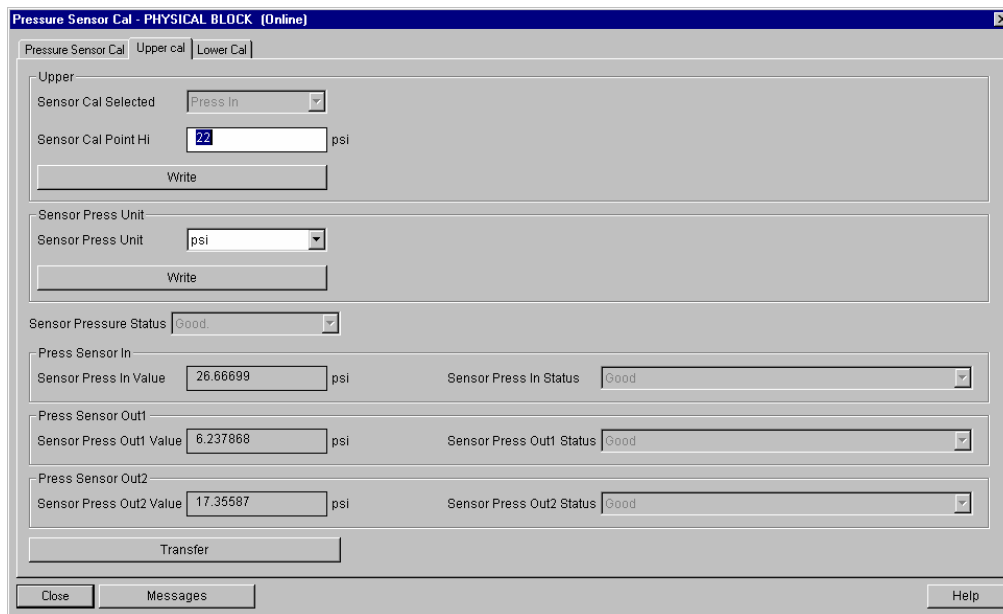


Figura 3 – Fácil visualização de status

Na figura 4 podemos ver o comportamento da posição real de uma válvula em função do SP.

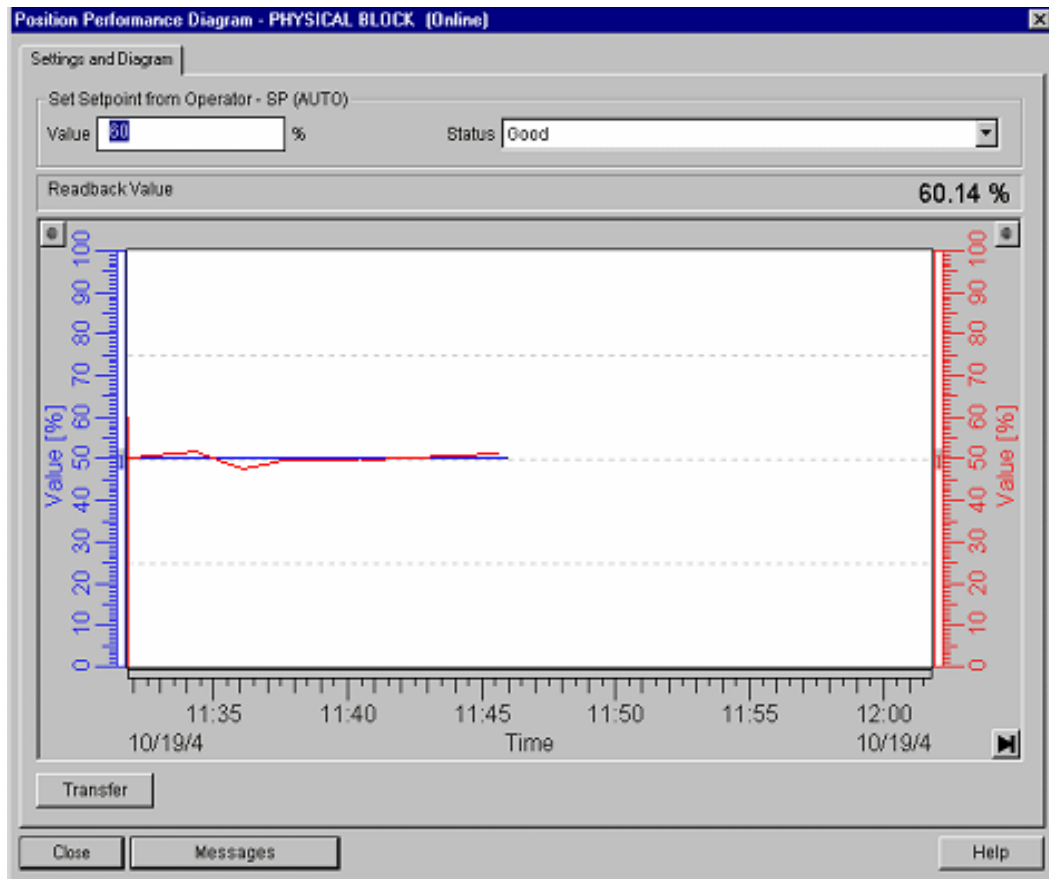


Figura 4 – Recursos de gráficos na EDDL do FY303-Smar

## Conclusão

A tendência natural tecnológica é o aperfeiçoamento e a coexistência de vários protocolos, cada um buscando explorar ao máximo seus pontos positivos e a EDDL vem para facilitar isto, colocando à disposição dos desenvolvedores de equipamentos e sistemas um maior poder expressivo para descrição e interpretação dos recursos e interfaces, sem abrir mão das suas características de simplicidade em relação à plataforma.

## Referências

- Specification for PROFIBUS Device Description and Device Integration, Volume 2: EDDL Specification".
- Material de treinamento Profibus – Smar, César Cassiolato
- [www.smar.com.br](http://www.smar.com.br) (faça o download das EDDLs Smar gratuitamente)