# 情報セキュリティ教育のための e ラーニング教材作成システム ELSEC の開発

川上 昌俊† 安田 浩‡ 佐々木 良一‡

†‡東京電機大学
101-8457 東京都千代田区神田錦町 2-2
†kawakami@isl.im.dendai.ac.jp
‡{yasuda, sasaki}@im.dendai.ac.jp

**あらまし** 情報セキュリティに関する脅威とその対策は日々変化していくものがあり，それらをユーザに教育するためには，それらの変化に合わせて教育内容も柔軟に変化させる必要がある．このため，著者らは拡張性と柔軟性が高い情報セキュリティ教育用eラーニングコンテンツを作成するためのELSECというシステムを開発した．ELSECは教育効果を高めるとされているアニメーションを利用したeラーニングコンテンツの作成を容易にし，そのコンテンツは多くのユーザがWeb上で快適に利用できる．本稿では，ELSECの開発目的，構成，機能等について記述した後，ELSECのフィッシング対策教育への適用の概要と評価について報告する．

# Development of an e-Learning Content-Making System for Information Security (ELSEC)

Masatoshi Kawakami† Hiroshi Yasuda‡ Ryoichi Sasaki‡

†‡Tokyo Denki University
2-2, Kanda Nisikicho Chiyoda-Ku, Tokyo 101-8457, Japan
†kawakami@isl.im.dendai.ac.jp
‡{yasuda, sasaki}@im.dendai.ac.jp

**Abstract** Since information security threats and their countermeasures change every day, an educational system must continuously be updated to reflect new threats and countermeasures. Therefore, to create scalable and flexible e-learning content for information security education, “ELSEC” was developed. ELSEC facilitates creating e-learning content with animations that enhance learning effect and the content can be used by many users on a website comfortably. This paper describes the development objectives, structure, and functions of ELSEC, as well as the summary of its application to anti-phishing education and evaluation.

## 1 はじめに

情報セキュリティに関する脅威には，マルウェアや不正アクセス，フィッシング(Phishing)など様々なものがある．そして，それらの脅威とその対策は日々変化するものが多い．

各ユーザがその対策を知り，活用する必要があるため，ユーザに対する情報セキュリティ教育は重要であると考えられる．また，脅威や対策が変化する場合，それに合わせて教育する内容も柔軟に変化させる必要がある．そのため，著者らは拡張性と柔軟性が高いeラーニングコンテンツを作成するための，ELSEC(E-Learning system for SECurity)というシステムを開発した．ELSECはこれ以外に，アニメーションを利用したコンテンツの作成の容易性や，そのコンテンツのWebサイトでの利用の快適性などの特徴を持っている．

本稿では，ELSECの開発目的，構成，機能等について記述した後，その評価を報告する．

## 2　ELSEC の開発

### 2.1　開発の背景

現在，多くの e ラーニングは「モチベーションの維持が困難」,「コンテンツがつまらない」,「効果が明瞭でない」等の問題を持っていると言われている[1]．そのような問題を改善するためには，教育活動の効果・効率・魅力を高めるための手法を集大成したモデルや研究分野である，インストラクショナルデザインを利用すると良いと考えられる[2]．

インストラクショナルデザインの研究では，熱中できるストーリーの一部としてアニメーションを用いることで e ラーニングの魅力が増すとされている[3]．また，ストーリーを導いてくれるエージェントと呼ばれるキャラクターがいることで学習効果が上がるとされている[4]．よって，適切にキャラクターが出てくるアニメーションを使用することで，e ラーニングの学習効果が上がると考えられる．

また，インストラクショナルデザインの一つに，Schank によって提唱された，行動することによって学ぶシナリオ型教材を設計するための理論である，ゴールベースシナリオ理論(Goal-Based Scenario 以下 GBS)というものがある[5]．これを利用することで，ユーザの動機づけを行い，効果的に学習目標を達成させることができると考えられる．GBS を利用して設計した教材は，シナリオを読み進めることや選択肢等によるシナリオ操作を必要とする．そのため，教材の適した実装方法の一つとして，アドベンチャーゲーム（Adventure Game 以下 AVG）形式の e ラーニングとしての実装が考えられる．

また，情報セキュリティに関する脅威や対策の変化が激しいため，それに柔軟に対応する必要がある．さらに，情報セキュリティ教育対象者は様々な人がおり，知識や学習に対する動機づけの強さといった特性がそれぞれ異なっている．そのため，各々の特性に合ったコンテンツを用意することにより学習効果が向上すると考えられる．従って，コンテンツの作成および変更が容易にできるシステムが必要である．

また，多くのユーザにコンテンツを利用してもらうために，様々な環境から Web サイト上で利用できるようにすると良いと考えられる．さらに，コンテンツのサイズが大きくてもダウンロード待ち時間が少なく，快適に利用できると，より利便性が高まる．

### 2.2　システムの要件

以上より，以下の要件を満たす情報セキュリティ教育用の e ラーニングコンテンツ作成システムが必要であると考えられる．

要件①　アニメーションを利用したコンテンツの容易な作成が可能である

要件②　AVG 形式のコンテンツの容易な作成と変更が可能である

要件③　Web サイト上で快適な利用ができるコンテンツが作成可能である

### 2.3　ELSEC の構成

著者らは，上記の要件を満たすシステムである ELSEC を開発した(図 1)．ELSEC では，要件①を満たすために Digital Movie Director(以下 DMD)を使用し，要件②，③を満たすために KScripter というスクリプトエンジンを開発し，使用している．それにより，拡張性と柔軟性があり，学習効果の高い e ラーニングコンテンツの作成を可能としている．

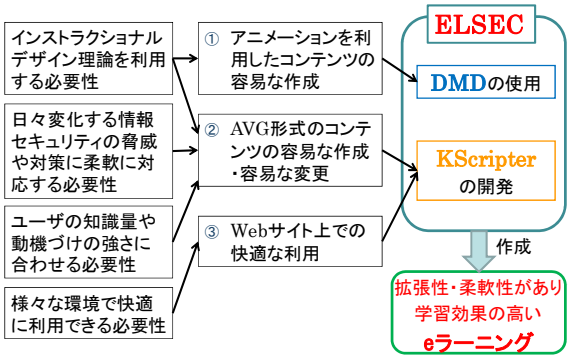


図 1．ELSEC の開発背景・要件・構成

### 2.4　DMD の使用

DMD とは，本論文の著者の一人である安田らが開発した 3 次元 CG アニメーション作成ソフトである．DMD では，主語，述語動詞，目的語等をリストから選択するだけで，容易にかつ短時間でアニメーションの作成が可能である．安田らの実験では，アニメーション作成スキルのない高校生でも DMD を使用することで，平均約 2 分間のアニメーションを平均 147 分間で作成できている[6]．

従って，DMD を使用することで，要件①を満たすことができると考えられる．

### 2.5 KScripter の開発

KScripter とは，ActionScript 3.0（以下 AS3）で開発した Flash ベースのスクリプトエンジンである．KScripter は AVG 形式の e ラーニングコンテンツの容易な作成と容易な修正を可能にすることを目的としている．そのため，平易な文法により AVG を作成できる NScripter という Windows 上で動作するスクリプトエンジン[7]の文法を参考に開発した．

#### 2.5.1 KScripter の機能

KScripter 自体は SWF 形式であり，テキスト形式で記述したスクリプトファイルを読み込み，解釈し，そのスクリプトに書かれたタイミングで画像・動画・音声ファイルを読み込み，文章や読み込んだ画像・動画等を表示・再生する機能を持っている（図 2）．KScripter で扱うことができるファイル形式は表 1 のとおりである．動画ファイルを扱うことができるため，DMD で作成したアニメーションを使用することも可能である．

また，KScripter は合計 47 個の命令を持っており，それらを使用することにより，選択肢の作成，画像の表示状態の切り替えや画像の移動による簡易アニメーションの作成，変数操作や条件分岐などもできる[8]．それらのうち，主要な 29 個の命令の引数，説明および使用例を表 3 に示す．

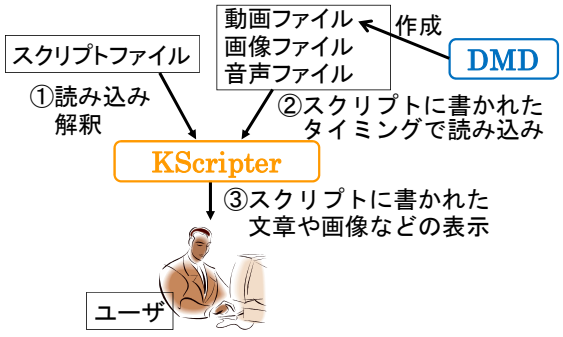


図 2. KScripter が持つ機能の概要

表 1. KScripter で扱えるファイル形式

| ファイル | 形式 |
|---|---|
| 画像ファイル | JPEG, PNG, GIF |
| 動画ファイル | FLV, SWF |
| 音声ファイル | MP3 |

さらに，KScripter は，過去の文章を再度表示するバックログ機能や文章を高速で読み進めるスキップ機能，自動で読み進めるオートリード機能，音声を消すミュート機能，文章の表示速度やオートリードの速度を調節する機能を持っている．そして，それらの機能に対応したボタンやスライダーが KScripter の実行画面の下部に表示されるようになっている（図 **3**）．

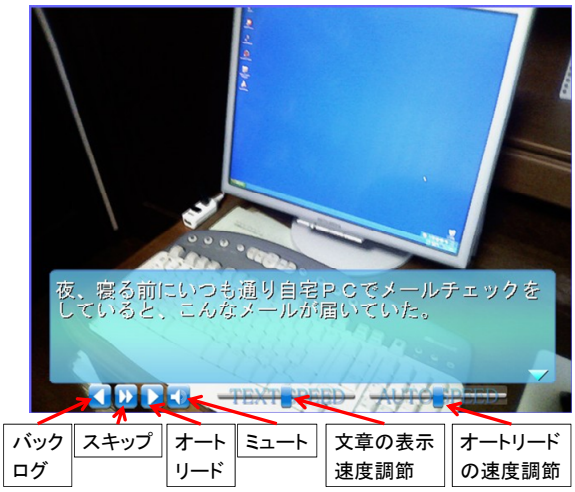


図 3. KScripter の実行画面にある機能

#### 2.5.2 KScripter の文法

KScripter は，NScripter に似た平易な文法を持っている．このためコンテンツの作成が非常に容易である．例えば，図 4 のような内容のスクリプトファイルを作成し，KScripter を実行すると，図 5 のように，画面下部のテキストウィンドウに「学習を開始しますか？」と表示され，画面上部に「する」と「しない」という選択肢が表示される．表示された選択肢から「する」を選んだ場合，「*start」以下が実行されるようになっているため，「それでは学習を開始します」と表示される．このように，文章の表示や選択肢の作成は非常に容易に行うことができる．

また，画像ファイルや音声ファイル，動画ファイルの読み込みや表示，再生なども，平易な文法によって実現することができる．

```
学習を開始しますか？
[select "する", *start, "しない", *end]
*start
それでは学習を開始します
……
*end
それではこれで終了します
```

図 4. KScripter 用のスクリプトの例

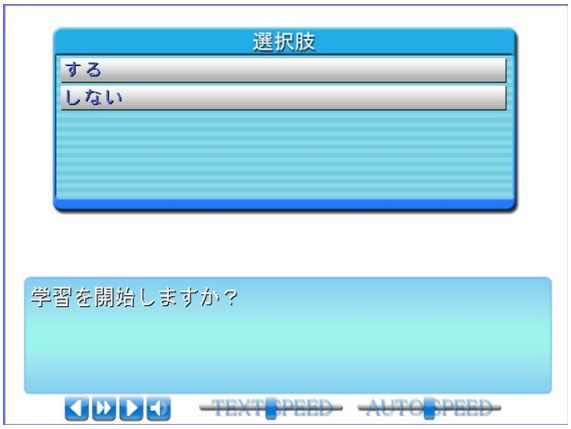

図 5. 図 4 のスクリプトの実行画面

ここで，ファイルの読み込みや表示などの操作を，一般的な記法で AS3 および KScripter によって書いた場合のそれぞれの論理 LOC（logical Lines Of Code）を表 2 に示す．表 2 から，KScripter の方が AS3 よりも少ない行数でファイル操作が可能であると言える．これは AS3 で複数行必要な処理を KScripter の一つの命令としてまとめているためである．また，ファイル操作以外の多くの命令も同様であるため，多くの処理は AS3 よりも KScripter の方が短い行数で書くことができる．

表 2. AS3 と KScripter のファイル操作に必要な論理 LOC の比較

| 操作 | AS3 | KScripter |
|---|---|---|
| 画像を読み込み，座標を設定して表示 | 12 行 | 2 行 |
| 動画を読み込み，座標を設定して表示・再生 | 12 行 | 3 行 |
| 音声を読み込み，再生 | 10 行 | 1 行 |

以上より，KScripter を使用することで，要件②を満たすことができると考えられる．

### 2.5.3 全体でのダウンロード待ち時間の短縮

KScripter によって作成された Web サイト上にあるコンテンツを表示するために，最初にダウンロードする必要があるファイルは，約 200KB の SWF ファイルとスクリプトが書かれたテキストファイルのみである．

また，使用する画像および動画ファイルはスクリプトで読み込み命令を記述した行でダウンロードを開始し，表示命令や再生命令を記述した行で表示や再生を行う．そのため，サイズが大きい画像や動画は，その表示や再生が必要になる行の数行から数十行前でダウンロードを開始することにより，表示や再生時の待ち時間を短縮することができる．

さらに，動画および音声ファイルの再生にはストリーミング再生を使用しているため，それによっても待ち時間は短縮される．

従って，要件③を満たすことができると考えられる．

## 3 評価

著者らは ELSEC を用いてフィッシング対策教育を行うための e ラーニングシステムを作成するという形で適用を行った[9]．その e ラーニングシステムは，主に以下の特徴を持たせることで，学習効果を高めている．

- GBS に基づいて設計した，AVG 形式のシナリオ型教材を主要コンテンツとする
- 複数のコンテンツを用意し，その中からユーザの事前知識量や学習への動機づけの強さに合ったコンテンツを勧める
- DMD によって作成した図 **6** に示すようなアニメーションを使用する
- シナリオの途中で，適宜確認テストを実施する

図 6. DMD で作成したアニメーション

このように様々な特徴を持つ複数のコンテンツの作成が必要であったが，ELSEC によって作成した結果，以下のようになった．

(1) KScripter による記述は，合計約 1900 ステップであり，KScripter の開発者一人で作成に要した時間は実質 30 日程

度であった．キャラクターの台詞や説明などの文章を除く，KScripter のコマンドのみのステップ数は約 1400 であった．また，作成の途中段階で，いろいろな人の意見を聞いて修正を行ったがその修正は容易であった．

(2) DMD によって作成した動画部は 11 場面（合計 12 分 20 秒）あったが，初めての適用で，約 6 時間で容易に作成できた．

(3) プログラムのダウンロードを開始し利用できるようになるまでに要する時間は，ダウンロード速度約 2. 3Mbps 以上で 1 秒以下である．

また，この e ラーニングの評価実験に参加したユーザの多くに十分な学習効果が現れ，e ラーニングの全体的な満足度などの評価も高かった．

一回の適用ではあるが，ELSEC は情報セキュリティ教育のための e ラーニングコンテンツ作成システムとして適切なものである見通しが得られた．

また，今回の適用によって判明した ELSEC の改善すべき点を以下に示す．

- ユーザインターフェースのヘルプがないため，ボタンやスライダー等の使い方がわかりづらい
- テキストウィンドウの色が明るすぎて，その上の白い文字が読みづらい
- データ保存機能がないため，途中で止めて，後でそこから再開するということができない
- アニメーションの会話のテンポが少し遅い

## 4 おわりに

本稿では，ELSEC の開発および評価について報告した．

ELSEC を使用したことにより，アニメーションが含まれ，学習効果や学習者の満足度等が高い e ラーニングシステムを容易に作成できた．しかし，適用が一回だけだったことや，その適用は ELSEC の一要素である KScripter の開発者が行ったものであるため，ELSEC による e ラーニング作成の容易性や利便性などの評価は不十分であると考えられる．また，今回の適用から ELSEC のいくつかの改善点も判明した．

今後は，ELSEC の改善点を対処していくとともに，開発者以外の人に ELSEC の適用を行ってもらうことにより，その評価を行っていきたい．

## 参考文献


[1] 経済産業省商務情報政策局情報処理振興課，”e ラーニング白書 2007/2008年版”，東京電機大学出版局，2007.

[2] 鈴木克明，”e-Learning実践のためのインストラクショナル・デザイン”，日本教育工学会論文誌, Vol.29, No.3, pp.197-205, 2005.

[3] Shank, R.C. "Lesson in learning, e-learning, and training", San Francisco: Pfeiffer, pp.222-223, 2005.

[4] Moreno, R., Mayer, R. E., Spires, H. A., and Lester, J. C. “The case for social agency in computer-based teaching: Do students learn more deeply when they interact with animated pedagogical agents?" Cognition and Instruction, Vol.19, No.2, pp.177-213, 2001.

[5] Schank, R. C. “Goal-Based Scenarios: Case-Based Reasoning Meets Learning by Doing”, Case-Based Reasoning: Experiences, Lessons & Future Directions (ed. D. B. Leake), AAAI Press/The MIT Press, pp. 295-347, 1996.

[6] 江村恒一，青樹輝勝，安田浩，”DMDシステムを用いた3次元アニメーション制作の評価”，情報処理学会研究報告．グラフィクスとCAD研究会報告, 2006(18) , pp.99-104，2006.

[7] 高橋直樹，“Takahashi's Web”, http://www.nscripter.com/

[8] “KScripter の命令一覧”, http://www.isl.im.dendai.ac.jp/kscripter/commands.html

[9] 川上昌俊，佐々木良一，”情報セキュリティ教育のためのeラーニング教材作成システム ELSEC のフィッシング対策教育への適用”，コンピュータセキュリティシンポジウム 2009(CSS2009), 予稿集, 2009.

表 3. KScripter の命令，引数，説明および使用例の一部

| 命令 | 引数 | | 説明 | 使用例 |
|---|---|---|---|---|
| | 型 | 意味 | | |
| **bg** | String,<br>uint,<br>uint = 0 | ファイル名,<br>エフェクト番号,<br>エフェクトにかける時間[ms] | 背景画像を読み込み，エフェクトをかけて表示する | [bg "bg1.jpg", 3, 1500] |
| **lsp** | uint,<br>String,<br>int,<br>int,<br>[uint] | スプライト番号,<br>ファイル名,<br>x座標,<br>y座標,<br>[不透明度(0～255) 省略時は255] | スプライト(画像)を表示状態で読み込む<br>(print命令で画面に反映) | [lsp 1, "image1.png", 30, 25] |
| **print** | uint,<br>uint = 0 | エフェクト番号,<br>エフェクトにかける時間[ms] | 指定エフェクトでスプライトを画面に反映する | [print 5, 2000] |
| **vsp** | int,<br>int | スプライト番号,<br>表示・非表示(1・0) | スプライトの表示・非表示を切り替える(print命令で画面に反映) | [vsp 1, 1] |
| **csp** | int | スプライト番号 | メモリ上からスプライトを消去する(スプライト番号を-1にすると全スプライト消去) | [csp 1] |
| **delay** | uint | 待ち時間[ms] | 指定時間だけ待つ(クリックでキャンセル可能) | [delay 1000] |
| **wait** | uint | 待ち時間[ms] | 指定時間だけ待つ(クリックでキャンセル不可) | [wait 1000] |
| **goto** | Label | ラベル名(*で始まる識別子) | 指定ラベルに飛ぶ | [goto *label1] |
| **bgm** | String,<br>uint | ファイル名,<br>フェードインにかける時間[ms] | MP3ファイルを読み込み，ループ再生する | [bgm "bgm1.mp3", 0] |
| **bgmstop** | uint | フェードアウトにかける時間[ms] | 再生中のMP3ファイルを停止する | [bgmstop 0] |
| **lswf** | uint,<br>String,<br>int,<br>int,<br>[uint] | スプライト番号,<br>ファイル名,<br>x座標,<br>y座標,<br>[不透明度(0～255) 省略時は255] | SWFを表示状態で読み込む(print命令で画面に反映) | [lswf 2, "movie1.swf", 0, 10] |
| **swfplay** | uint | スプライト番号 | SWFファイルのタイムライン内で再生ヘッドを移動する | [swfplay 2] |
| **swfstop** | uint | スプライト番号 | SWFファイルの再生ヘッドを停止する | [swfstop 2] |
| **lflv** | uint,<br>String,<br>int,<br>int,<br>[uint] | スプライト番号,<br>ファイル名,<br>x座標,<br>y座標,<br>[不透明度(0～255) 省略時は255] | FLVを表示状態で読み込む(print命令で画面に反映) | [lflv 2, "movie1.flv", 0, 10] |
| **flvplay** | uint | スプライト番号 | 指定スプライト番号のFLVを再生する | [flvplay 2] |
| **wflvc** | uint | スプライト番号 | 指定スプライト番号のFLVの再生が終了するまで待つ(指定スプライト番号のFLVが存在しない場合,何もしない) | [wflvc 2] |
| **textoff** | void | なし | テキストウィンドウを一時消去する(文章の表示があると自動的に表示される) | [textoff] |
| **texton** | void | なし | テキストウィンドウを表示する | [texton] |
| **click** | void | なし | クリックするまで待つ | [click] |
| **select** | String,<br>Label<br>[,String,<br>Label] | 表示する文字列,<br>ラベル名,<br>[(選択肢の数だけ繰り返し)] | 選択肢を表示する．ユーザが選択した選択肢の直後の引数となっているラベルに飛ぶ． | [select "選択肢1", *label1, "選択肢2", *label2] |
| **mov** | $var,<br>Object | $変数名,<br>数値or文字列 | $varにObjectを代入する | [mov $str, "文字列"] |
| **add** | $var,<br>Object | $変数名,<br>数値or文字列 | $varとObjectの和を$varに代入する(どちらかが文字列の場合，結合) | [add $num1, 10] |
| **sub** | $var,<br>Number | $数値型変数名,<br>数値 | $varとNumberの差を$varに代入する | [sub $num1, 1.23] |
| **mul** | $var,<br>Number | $数値型変数名,<br>数値 | $varとNumberの積を$varに代入する | [mul $num1, -5] |
| **div** | $var,<br>Number | $数値型変数名,<br>数値 | $varとObjectの商を$varに代入する | [div $num1, 2] |
| **if** | 条件文 | 条件文 | 必ず[endif]とセットで使用し，条件文が真ならば[endif]までのスクリプトを実行し，偽ならば[endif]まで実行しない | [if $num1 < 0 && $num2 != -1]<br>条件文が真ならば，ここを実行．<br>[endif] |
| **endif** | void | なし | ifの説明を参照 | ifの使用例を参照 |
| **msp** | uint,<br>int,<br>int,<br>[int] | スプライト番号,<br>加算x座標,<br>加算y座標,<br>[加算不透明度 省略時は無変化] | 指定スプライトの座標と不透明度に指定数値だけ加算する(print命令を使わなくても,この命令実行時に移動が反映される) | [msp 10, 50, -50, -100] |
| **spmt** | uint,<br>int,<br>int,<br>uint,<br>uint | スプライト番号,<br>移動先x座標,<br>移動先y座標,<br>不透明度(0～255),<br>移動時間[ms] | 指定スプライトを指定座標&指定不透明度(0～255)に指定時間で移動させる | [spmt 2, 100, 200, 255, 1500] |