Towards Algebraic High-Level Systems as Weak Adhesive HLR Categories

Ulrike Prange[1,2]

Department of Software Engineering and Theoretical Computer Science
Technical University of Berlin
Berlin, Germany

Abstract

Adhesive high-level replacement (HLR) systems have been recently established as a suitable categorical framework for double pushout transformations based on weak adhesive HLR categories. Among different types of graphs and graph-like structures, various kinds of Petri nets and algebraic high-level (AHL) nets are interesting instantiations of adhesive HLR systems. AHL nets combine algebraic specifications with Petri nets to allow the modeling of data, data flow and data changes within the net.
For the development and analysis of reconfigurable systems, not only AHL schemas based on an algebraic specification and AHL nets using an additional algebra should be considered, but also AHL systems which additionally include markings of nets.
In this paper, we summarize the results for different kinds of AHL schemas and nets, and extend these results to AHL systems. The category of markings is introduced, which allows a general construction combining AHL nets with possible markings leading under certain properties to a weak adhesive HLR category.

Keywords: Algebraic High-Level Nets, Algebraic High-Level Systems, Adhesive HLR Categories

1 Introduction

Petri nets are an important modeling technique to describe discrete distributed systems. Their nondeterministic firing steps are well-suited for modeling the concurrent behavior of such systems.

As the adaptation of a system to a changing environment gets more and more important, Petri nets that can be transformed during runtime have become a significant topic in the recent years. Application areas cover e.g. computer supported cooperative work, multi agent systems, dynamic process mining and mobile networks. Moreover, this approach increases the expressiveness of Petri nets and allows a formal description of dynamic changes.

For the terminology in this paper, a Petri net describes only the structure of the net, while a Petri system consists of a Petri net and a suitable marking. In

[1] This work was partially funded by the research project forMA$_\ell$NET of the German Research Council (see http://tfs.cs.tu-berlin.de/formalnet/).

[2] Email: uprange@cs.tu-berlin.de

[6], the concept of reconfigurable place/transition (P/T) systems was introduced for modeling changes of the net structure while the system is kept running. In detail, a reconfigurable P/T system consists of a P/T system and a set of rules, so that not only the follower marking can be computed but also the net structure can be changed by rule application. So, a new P/T system is obtained that is more appropriate with respect to some requirements of the environment.

As an extension of Petri nets, algebraic high-level (AHL) nets combine algebraic specifications with P/T nets [8] to allow the modeling of data, data flow and data changes within the net. In general, an AHL system denotes an AHL net based on a specification SP in combination with an SP-algebra A and an initial marking M. Combining AHL systems with rules leads to reconfigurable AHL systems.

In this paper, we integrate rule-based transformations of AHL systems into the framework of adhesive high-level replacement (HLR) systems [2,4] that is inspired by graph transformation systems. Adhesive HLR systems are a suitable categorical framework for graph transformation in the double pushout approach. They combine the framework of HLR systems [3] with the framework of adhesive categories [7]. The main concept behind adhesive categories are the so-called van Kampen squares. These ensure that pushouts along monomorphisms are stable under pullbacks and, vice versa, that pullbacks are stable under combined pushouts and pullbacks. In the case of weak adhesive HLR categories, the class of all monomorphisms is replaced by a subclass $\mathcal{M}$ of monomorphisms closed under composition and decomposition, and in the van Kampen squares certain $\mathcal{M}$-morphisms are required.

The framework of weak adhesive HLR categories is sufficient to show under some additional assumptions as main results the Local Church-Rosser Theorem, the Parallelism Theorem, the Concurrency Theorem, the Embedding and Extension Theorem, and the Local Confluence Theorem, also called Critical Pair Lemma.

For different kinds of Petri nets we already know that the corresponding categories are weak adhesive HLR categories. For elementary nets, P/T nets and AHL schemas with a fixed specification this has been shown in [2], for AHL schemas and nets with suitable algebras in [9], and for P/T systems in [10]. The proof for P/T systems has been done by showing directly the different properties of a weak adhesive HLR category for P/T nets with markings. Analogously, this could be done for each kind of AHL system. But the more elegant way is to show that there is a categorical construction combining AHL nets and their markings, leading to a general proof for different kinds of low-level and high-level Petri net systems. Therefore we introduce the category **Markings** of markings and show that AHL systems can be considered as a comma category of AHL nets and markings, leading to a weak adhesive HLR category if the underlying category of AHL nets is a weak adhesive HLR category with suitable algebras.

This paper is organized as follows: In Section 2, we introduce the basic notions of weak adhesive HLR categories and adhesive HLR systems. Known results for P/T nets, AHL schemas, AHL nets and P/T systems are summarized in Section 3. In Section 4, the category of markings is defined leading to the weak adhesive HLR category of AHL systems in Section 5. A small example of a reconfigurable AHL system is presented in Section 6. At last, in Section 7 the conclusion is given and future work is described.

2 Weak Adhesive HLR Categories and Adhesive HLR Systems

In this section, we introduce weak adhesive HLR categories and adhesive HLR systems. For a more detailed view we refer to [2].

The intuitive idea of weak adhesive HLR categories are categories with suitable pushouts and pullbacks which are compatible with each other. More precisely the definition is based on so-called van Kampen squares.

The idea of a van Kampen square is that of a pushout being stable under pullbacks, and vice versa that pullbacks are stable under combined pushouts and pullbacks.

Definition 2.1 (Van Kampen square) A pushout (1) is a *van Kampen square*, if for any commutative cube (2) with (1) in the bottom and the back faces being pullbacks holds: the top face is a pushout if and only if the front faces are pullbacks.

$$\begin{tikzcd} A \arrow[r, "m"] \arrow[d, "f"'] \arrow[dr, phantom, "(1)"] & B \arrow[d, "g"] \\ C \arrow[r, "n"'] & D \end{tikzcd}
\qquad
\begin{tikzcd} & & A' \arrow[dll, "f'"'] \arrow[drr, "m'"] \arrow[ddd, "a"'] & & \\ C' \arrow[dr, "n'"] \arrow[ddd, "c"] & & & & B' \arrow[dlll, "g'"'] \arrow[ddd, "b"'] \\ & D' \arrow[ddd, "d"] & & & \\ & & A \arrow[dll, "f"'] \arrow[drr, "m"] & & \\ C \arrow[dr, "n"'] & & & & B \arrow[dlll, "g"] \\ (2) & D & & & \end{tikzcd}$$

Since not even in the category **Sets** of sets and functions each pushout is a van Kampen square, for weak adhesive HLR categories only those squares are considered where m is an $\mathcal{M}$-morphism, and some more morphisms in the cube are required to be in $\mathcal{M}$.

Definition 2.2 (Weak adhesive HLR category) A category $\mathbf{C}$ with a morphism class $\mathcal{M}$ is a *weak adhesive HLR category*, if

(i) $\mathcal{M}$ is a class of monomorphisms closed under isomorphisms, composition ($f : A \to B \in \mathcal{M}, g : B \to C \in \mathcal{M} \Rightarrow g \circ f \in \mathcal{M}$) and decomposition ($g \circ f \in \mathcal{M}, g \in \mathcal{M} \Rightarrow f \in \mathcal{M}$),

(ii) $\mathbf{C}$ has pushouts and pullbacks along $\mathcal{M}$-morphisms and $\mathcal{M}$-morphisms are closed under pushouts and pullbacks,

(iii) pushouts in $\mathbf{C}$ along $\mathcal{M}$-morphisms are weak VK squares.

For a weak VK square, the VK square property holds for all commutative cubes with $m \in \mathcal{M}$ and ($f \in \mathcal{M}$ or $b, c, d \in \mathcal{M}$) (see Def. 2.1).

For historical reasons, these categories are called weak adhesive HLR categories. In [11] and related work, adhesive categories are used as the categorical framework for deriving process congruences from reaction rules. The step from adhesive to adhesive HLR categories is justified by the fact that there are some important examples – such as algebraic specifications and typed attributed graphs – which are not adhesive categories. However, they are adhesive HLR categories for a suitable subclass $\mathcal{M}$ of all monomorphisms. Thus, the main difference between adhesive HLR categories and adhesive categories is that a distinguished class $\mathcal{M}$ of monomorphisms is considered instead of all monomorphisms, so that only pushouts along $\mathcal{M}$-morphisms have to be VK squares. Another important example – the category **PTNets** of place/transition nets with the class $\mathcal{M}$ of injective morphisms – fails to be an adhesive HLR category, but is a weak adhesive HLR category with the restriction to weak van Kampen squares. This justifies the step to weak adhesive HLR categories.

The categories **Sets** of sets and functions, **Graphs** of graphs and graph morphisms, and $\mathbf{Graphs_{TG}}$ of typed graphs and typed graph morphisms are weak adhesive HLR categories for the class $\mathcal{M}$ of all monomorphisms. Moreover, an important example is the category $(\mathbf{AGraphs_{ATG}}, \mathcal{M})$ of typed attributed graphs with a type graph ATG and the class $\mathcal{M}$ of all injective morphisms with isomorphisms on the data part.

Weak adhesive HLR categories are closed under product, slice, coslice, functor, and comma category constructions. This means that we can construct new weak adhesive HLR categories from given ones [2,9].

Theorem 2.3 (Construction Theorem) *If $(\mathbf{C}, \mathcal{M}_1)$ and $(\mathbf{D}, \mathcal{M}_2)$ are weak adhesive HLR categories, then the following categories are also weak adhesive HLR categories:*

(i) *the full subcategory $(\mathbf{C}', \mathcal{M}')$ of $\mathbf{C}$ with $\mathcal{M}' = \mathcal{M}_1|_{\mathbf{C}'}$ if $\mathbf{C}'$ has pushouts and pullbacks along $\mathcal{M}'$-morphisms which are preserved by the inclusion functor,*

(ii) *the product category $(\mathbf{C} \times \mathbf{D}, \mathcal{M}_1 \times \mathcal{M}_2)$,*

(iii) *the slice category $(\mathbf{C}\backslash X, \mathcal{M}_1 \cap \mathbf{C}\backslash X)$ and the coslice category $(X\backslash\mathbf{C}, \mathcal{M}_1 \cap X\backslash\mathbf{C})$ for any object X in $\mathbf{C}$,*

(iv) *for every category $\mathbf{X}$ the functor category $([\mathbf{X}, \mathbf{C}], \mathcal{M}_1$-functor transformations), where an $\mathcal{M}_1$-functor transformation is a natural transformation $t : F \to G$ where all morphisms $t_X : F(X) \to G(X)$ are in $\mathcal{M}_1$,*

(v) *the comma category $(ComCat(F, G; \mathcal{I}), \mathcal{M})$ with $\mathcal{M} = (\mathcal{M}_1 \times \mathcal{M}_2) \cap Mor_{ComCat(F,G;\mathcal{I})}$ and functors $F : \mathbf{C} \to \mathbf{X}$, $G : \mathbf{D} \to \mathbf{X}$, where F preserves pushouts along $\mathcal{M}_1$-morphisms and G preserves pullbacks along $\mathcal{M}_2$-morphisms.*

Now we are able to generalize graph transformation systems, grammars and languages in the sense of [1,2].

In general, an adhesive HLR system is based on productions, also called rules, that describe in an abstract way how objects in this system can be transformed. An application of a production is called a direct transformation and describes how an object is actually changed by the production. A sequence of these applications yields a transformation.

Definition 2.4 (Production and transformation) Given a weak adhesive HLR category $(\mathbf{C}, \mathcal{M})$, a *production* $p = (L \xleftarrow{l} K \xrightarrow{r} R)$ (also called rule) consists of three objects L, K and R called left hand side, gluing object and right hand side respectively, and morphisms $l : K \to L$, $r : K \to R$ with $l, r \in \mathcal{M}$.

Given a production $p = (L \xleftarrow{l} K \xrightarrow{r} R)$ and an object G with a morphism $m : L \to G$, called match, a *direct transformation* $G \xRightarrow{p,m} H$ from G to an object H is given by the following diagram, where (1) and (2) are pushouts. A sequence $G_0 \Rightarrow G_1 \Rightarrow ... \Rightarrow G_n$ of direct transformations is called a *transformation* and is denoted as $G_0 \overset{*}{\Rightarrow} G_n$.

$$\begin{tikzcd}
L \arrow[d, "m"'] \arrow[dr, phantom, "(1)"] & K \arrow[l, "l"'] \arrow[r, "r"] \arrow[d, "k"'] \arrow[dr, phantom, "(2)"] & R \arrow[d, "n"] \\
G & D \arrow[l, "f"] \arrow[r, "g"'] & H
\end{tikzcd}$$

Definition 2.5 (Adhesive HLR system, grammar and language) An *adhesive HLR system* $AHS = (\mathbf{C}, \mathcal{M}, P)$ consists of a weak adhesive HLR category $(\mathbf{C}, \mathcal{M})$ and a set of productions P.

An *adhesive HLR grammar* $AHG = (AHS, S)$ is an adhesive HLR system AHS together with a distinguished start object S.

The *language* L of an adhesive HLR grammar $AHG = (AHS, S)$ is defined by

$$L = \{G \mid \exists \text{ transformation } S \overset{*}{\Rightarrow} G\}.$$

Note that there are two different kinds of systems: adhesive HLR systems, which consist of a weak adhesive HLR category and some productions, and Petri systems, which consist of a Petri net together with a suitable marking. Instantiating adhesive HLR systems with Petri systems leads to reconfigurable Petri systems, i.e. Petri nets with markings and productions.

For the theory of adhesive HLR systems we refer to [2]. In the following, we only analyze which kinds of low-level and high-level Petri nets are weak adhesive HLR categories. To apply the whole general theory of adhesive HLR systems to these nets, some more properties are necessary which are not handled here.

3 P/T Nets, AHL Schemas, AHL Nets and P/T Systems as Weak Adhesive HLR Categories

In this section, we review under which conditions different kinds of P/T and AHL schemas, nets and systems are weak adhesive HLR categories. We only define the structures and present the results. The corresponding proofs can be found in [2,9,10].

Definition 3.1 An *elementary net* is given by $EN = (P, T, pre, post)$ with sets P of places, T of transitions, and pre- and post-domain functions $pre, post : T \to \mathcal{P}(P)$, where $\mathcal{P}$ is the power set functor.

An *elementary net morphism* $f_{EN} : EN \to EN'$ is given by $f_{EN} = (f_P : P \to P', f_T : T \to T')$ compatible with the pre- and post-domain functions, i.e. $pre' \circ f_T = \mathcal{P}(f_P) \circ pre$ and $post' \circ f_T = \mathcal{P}(f_P) \circ post$.

Elementary nets and elementary net morphisms form the category **ElemNets**.

Corollary 3.2 *The category* $(\mathbf{ElemNets}, \mathcal{M})$ *is a weak adhesive HLR category, where* $\mathcal{M}$ *is the class of all injective morphisms [2,9].*

Note, that $(\mathbf{ElemNets}, \mathcal{M})$ is not an adhesive HLR category as stated in [2], since the power set functor $\mathcal{P}$ only preserves pullbacks along injective morphisms, but not over general ones.

Definition 3.3 A *place/transition net* $PN = (P, T, pre, post)$ is given by a set P of places, a set T of transitions, as well as pre- and post-domain functions $pre, post : T \to P^{\oplus}$, where $P^{\oplus}$ is the free commutative monoid over P.

A *place/transition net morphism* $f_{PN} : PN \to PN'$ is given by $f_{PN} = (f_P : P \to P', f_T : T \to T')$ compatible with the pre- and post-domain functions, i.e. $pre' \circ f_T = f_P^{\oplus} \circ pre$ and $post' \circ f_T = f_P^{\oplus} \circ post$.

Place/transition nets and place/transition net morphisms form the category **PTNets**.

Corollary 3.4 *The category* $(\mathbf{PTNets}, \mathcal{M})$ *is a weak adhesive HLR category, where* $\mathcal{M}$ *is the class of all injective morphisms [2].*

Definition 3.5 A *place/transition system* $PS = (PN, m)$ is given by a place/transition net $PN = (P, T, pre, post)$ and a marking $m : P \to \mathbb{N}$.

A *place/transition system morphism* $f_{PS} : PS \to PS'$ is given by a place/transition net morphism $f_{PN} : PN \to PN'$ that is marking-preserving, i.e. $\forall p \in P : m(p) \le m'(f_{PN}(p))$.

Place/transition systems and place/transition system morphisms form the category **PTSystems**.

Corollary 3.6 *The category* $(\mathbf{PTSystems}, \mathcal{M})$ *is a weak adhesive HLR category, where* $\mathcal{M}$ *is the class of all strict morphisms, i.e.* $f_{PS} : PS \to PS' \in \mathcal{M}$ *if* f_{PN} *is injective and marking-strict:* $\forall p \in P : m(p) = m'(f_{PN}(p))$ *[10].*

Definition 3.7 An *AHL schema* over an algebraic specification SP, where $SP = (SIG, E, X)$ has additional variables X and $SIG = (S, OP)$, is given by $AC = (P, T, pre, post, cond, type)$ with sets P of places and T of transitions, $pre, post : T \to (T_{SIG}(X) \otimes P)^{\oplus}$ as pre- and post-domain functions, $cond : T \to \mathcal{P}_{fin}(Eqns(SIG, X))$ assigning to each $t \in T$ a finite set $cond(t)$ of equations over SIG and X, and $type : P \to S$ a type function. Note that $T_{SIG}(X)$ is the SIG-term algebra with variables X and $(T_{SIG}(X) \otimes P) = \{(term, p) \mid term \in T_{SIG}(X)_{type(p)}, p \in P\}$.

An *AHL schema morphism* $f_{AC} : AC \to AC'$ is given by a pair of functions $f_{AC} = (f_P : P \to P', f_T : T \to T')$ which are compatible with pre, $post$, $cond$ and $type$ as shown below.

$$\begin{tikzcd}
 & T \arrow[dl, "cond"'] \arrow[r, shift left, "pre"] \arrow[r, shift right, "post"'] \arrow[dd, "f_T"] & (T_{SIG}(X) \otimes P)^{\oplus} \arrow[dd, "(id \otimes f_P)^{\oplus}"] & P \arrow[dr, "type"] \arrow[dd, "f_P"'] & \\
\mathcal{P}_{fin}(Eqns(SIG, X)) = & = & & = & S \\
 & T' \arrow[ul, "cond'"] \arrow[r, shift left, "pre'"] \arrow[r, shift right, "post'"'] & (T_{SIG}(X) \otimes P')^{\oplus} & P' \arrow[ur, "type'"'] &
\end{tikzcd}$$

Given an algebraic specification SP, AHL schemas over SP and AHL schema morphisms form the category $\mathbf{AHLSchemas(SP)}$.

Corollary 3.8 *The category* $(\mathbf{AHLSchemas(SP)}, \mathcal{M})$ *is a weak adhesive HLR category, where* $\mathcal{M}$ *is the class of all injective morphisms [2,9].*

Definition 3.9 An *AHL net* $AN = (AC, A)$ is given by an AHL schema AC over SP and an SP-algebra $A \in \mathbf{A(SP)}$, where $\mathbf{A(SP)}$ is a subcategory of $\mathbf{Algs(SP)}$, the category of all algebras over SP.

An *AHL net morphism* $f_{AN} : AN \to AN'$ is given by a pair $f_{AN} = (f_{AC} : AC \to AC', f_A : A \to A')$, where f_{AC} is an AHL schema morphism and $f_A \in \mathbf{A(SP)}$ an SP-homomorphism.

Given an algebraic specification SP, AHL nets over SP and AHL net morphisms form the category $\mathbf{AHLNets(SP)}$.

Corollary 3.10 *If* $(\mathbf{A(SP)}, \mathcal{M})$ *is a weak adhesive HLR category then the category* $(\mathbf{AHLNets(SP)}, \mathcal{M}')$ *is a weak adhesive HLR category, where* $\mathcal{M}'$ *is the class of all morphisms* $f_{AN} = (f_{AC}, f_A)$ *with* f_{AC} *being injective and* $f_A \in \mathcal{M}$ *[9].*

For the algebra part, up to now it is not clear whether or under what conditions the category $\mathbf{Algs(SP)}$ of algebras over an arbitrary specification SP with the class $\mathcal{M}_{inj}$ of injective morphisms is a weak adhesive HLR category. Nevertheless, there are two possible choices for the category $(\mathbf{A(SP)}, \mathcal{M})$:

(i) The category $(\mathbf{Algs(SP)}, \mathcal{M}_{iso})$ with the class $\mathcal{M}_{iso}$ of isomorphisms, which is useful for systems where only the net part but not the algebra part is allowed to be changed by rule application.

(ii) The category $(\mathbf{Algs(SP)}, \mathcal{M}_{inj})$ with the class $\mathcal{M}_{inj}$ of injective morphisms, where SP is a graph structure algebra, which means that only unary operations are allowed.

Definition 3.11 A *generalized AHL schema* $GC = (SP, AC)$ is given by an algebraic specification SP and an AHL schema AC over SP.

A *generalized AHL schema morphism* $f : GC \to GC'$ is a tuple $f_{GC} = (f_{SP} : SP \to SP', f_P : P \to P', f_T : T \to T')$, where f_{SP} is a specification morphism and f_P, f_T are compatible with *pre*, *post*, *cond* and *type*. $f^{\#}_{SP}$ is the extension of f_{SP} to terms and equations.

$$
\begin{tikzcd}
\mathcal{P}_{fin}(Eqns(SIG, X)) \arrow[d, "\mathcal{P}_{fin}(f^{\#}_{SP})"'] \arrow[r, phantom, "="] & T \arrow[l, "cond"'] \arrow[r, shift left, "pre"] \arrow[r, shift right, "post"'] \arrow[d, "f_T"] \arrow[r, phantom, "=", shift right=6] & (T_{SIG}(X) \otimes P)^{\oplus} \arrow[d, "(f^{\#}_{SP} \otimes f_P)^{\oplus}"] & P \arrow[r, "type"] \arrow[d, "f_P"'] \arrow[r, phantom, "=", shift right=6] & S \arrow[d, "f_{SP,S}"] \\
\mathcal{P}_{fin}(Eqns(SIG', X')) & T' \arrow[l, "cond'"] \arrow[r, shift left, "pre'"] \arrow[r, shift right, "post'"'] & (T_{SIG'}(X') \otimes P')^{\oplus} & P' \arrow[r, "type'"'] & S'
\end{tikzcd}
$$

Generalized AHL schemas and generalized AHL schema morphisms form the category $\mathbf{AHLSchemas}$.

Corollary 3.12 *The category* $(\mathbf{AHLSchemas}, \mathcal{M})$ *is a weak adhesive HLR category, where* $\mathcal{M}$ *is the class of all morphisms* $f_{GC} = (f_{SP}, f_P, f_T)$ *with* f_{SP} *being strict injective and* f_P, f_T *being injective [9].*

For the definition of generalized AHL nets we need the category **Algs** of algebras and generalized algebra homomorphisms, which is defined by

- algebras $A \in \mathbf{Algs(SP)}$ for a specification SP as objects,
- as morphisms, generalized algebra homomorphisms $f : A \to A'$ between algebras $A \in \mathbf{Algs(SP)}$ and $A' \in \mathbf{Algs(SP')}$, i.e. algebra homomorphisms $f : A \to V_h(A')$ in $\mathbf{Algs(SP)}$ for a specification morphism $h : SP \to SP'$, where $V_h : \mathbf{Algs(SP')} \to \mathbf{Algs(SP)}$ is the forgetful functor between the algebras.

Definition 3.13 A *generalized AHL net* $GN = (GC, A)$ is given by a generalized AHL schema GC over the algebraic specification SP and an SP-algebra $A \in \mathbf{A}$, where $\mathbf{A}$ is a subcategory of **Algs**.

A *generalized AHL net morphism* $f_{GN} : GN \to GN'$ is a tuple $f_{GN} = (f_{GC} : GC \to GC', f_{GA} : A \to V_{f_{SP}}(A'))$, where $f_{GC} = (f_{SP}, f_P, f_T)$ is a generalized AHL schema morphism and $f_{GA} \in \mathbf{A}$ a generalized algebra homomorphism. $V_{f_{SP}} : \mathbf{Algs(SP')} \to \mathbf{Algs(SP)}$ is the forgetful functor induced by f_{SP}.

Generalized AHL nets and generalized AHL net morphisms form the category **AHLNets**.

Corollary 3.14 *If* $(\mathbf{A}, \mathcal{M}_1)$ *is a weak adhesive HLR category of algebras, then the category* $(\mathbf{AHLNets}, \mathcal{M})$ *is a weak adhesive HLR category, where* $\mathcal{M}$ *is the class of all net morphisms* $f_{GN} = (f_{GC}, f_{GA})$ *with* f_{GC} *being strict injective and* $f_{GA} \in \mathcal{M}_1$ *[9].*

As in the case of AHL nets, up to now we do not know whether the category **Algs** with the class $\mathcal{M}_{inj}$ of injective morphisms is a weak adhesive HLR category and can be used for the algebra part. Again, we have two possible choices for the category $(\mathbf{A}, \mathcal{M}_\mathbf{1})$:

(i) The category $(\mathbf{Algs}, \mathcal{M}_{iso})$ with the class $\mathcal{M}_{iso}$ of isomorphisms, which is useful for systems where only the net part but not the algebra part is allowed to be changed by rule application.

(ii) The category $(\mathbf{Algs}|_{\mathbf{QTA}}, \mathcal{M}_{sinj})$ of quotient term algebras and unique induced homomorphisms, with the class $\mathcal{M}_{sinj}$ of strict injective morphisms.

4 The Category of Markings

In this section, we define the category **Markings** of markings and show that this category is a weak adhesive HLR category.

In general, a marking of a net can be seen as a multiset, i.e. an element of a free commutative monoid – in the case of P/T nets of $P^\oplus$, in the case of AHL nets of $(A \otimes P)^\oplus$, where $\otimes$ means the type-correct product. As a consequence, we could use the category **FCMonoids** of free commutative monoids for our markings. Unfortunately, in many cases the morphisms between P/T or AHL systems should not be marking-strict, which means that the marking on each place p has to be equal in both nets, as is the case for morphisms in **FCMonoids**.

For this reason, we define the category **Markings**, where the objects are sets combined with a function to natural numbers defining the quantity of each element of the set. For morphisms, we only require a mapping between the sets that preserves these quantities.

Definition 4.1 (Category Markings) The category **Markings** consists of

- objects (S, s) with a set S and a function $s : S \to \mathbb{N}$,
- morphisms $f : (S, s) \to (T, t)$ with a function $f : S \to T$ such that $\forall s_1 \in S : s(s_1) \leq t(f(s_1))$,
- a composition $g \circ f$ of $f : (S, s) \to (T, t)$, $g : (T, t) \to (U, u)$ with $\forall s_i \in S : g \circ f(s_1) = g(f(s_1))$ as in **Sets**,
- identities $id_{(S,s)} : (S, s) \to (S, s)$ with $id_{(S,s)} = id_S$ as in **Sets**.

This category is well-defined since the morphisms are basically morphisms in **Sets**, and for the composition we have $\forall s_1 \in S : s(s_1) \leq t(f(s_1)) \leq u(g(f(s_1)))$, which means $g \circ f$ is a valid **Markings**-morphism.

Now we shall show that the category of markings with a suitable morphism class $\mathcal{M}_{strict}$ of strict morphisms is a weak adhesive HLR category. First we define this morphism class $\mathcal{M}_{strict}$, and then we prove some lemmas which are necessary to show the desired result.

Definition 4.2 (strict morphism) A morphism $f : (S, s) \to (T, t)$ in **Markings** is marking-strict if $\forall s_1 \in S : s(s_1) = t(f(s_1))$.

A morphism $f : (S, s) \to (T, t)$ in **Markings** is strict, if f is injective and marking-strict. All strict morphisms form the morphism class $\mathcal{M}_{strict}$.

The category **FCMonoids** of free commutative monoids is a subcategory of **Markings**, where the morphisms in **FCMonoids** are exactly the marking-strict morphisms.

Lemma 4.3 *$\mathcal{M}_{strict}$ is a class of monomorphisms closed under composition and decomposition.*

Proof. Given morphisms $f : (S, s) \to (T, t)$, $g : (T, t) \to (U, u)$ in **Markings**, we have:

(i) If f is strict, then it is injective and we inherit from **Sets** that it is a monomorphism.

(ii) Injective morphisms in **Sets** are closed under composition and decomposition. This holds also in **Markings**.

(iii) If f, g are strict we have $\forall s_1 \in S : s(s_1) \overset{f \text{ strict}}{=} t(f(s_1)) \overset{g \text{ strict}}{=} u(g(f(s_1)))$, which means that also $g \circ f$ is strict.

(iv) If g, $g \circ f$ are strict we have $\forall s_1 \in S : s(s_1) \overset{g \circ f \text{ strict}}{=} u(g(f(s_1))) \overset{g \text{ strict}}{=} t(f(s_1))$, which means that also f is strict.

$\square$

The next proofs are very similar to the proofs for P/T systems being a weak adhesive HLR category in [10]. We generalize these proofs to the category of markings. First we shall show that pushouts along $\mathcal{M}_{strict}$-morphisms exist and preserve $\mathcal{M}_{strict}$-morphisms.

Lemma 4.4 *In* **Markings***, pushouts along $\mathcal{M}_{strict}$-morphisms exist and preserve $\mathcal{M}_{strict}$, i.e. given morphisms f and m with m strict, then the pushout (PO) exists and n is also a strict morphism.*

$$\begin{tikzcd}
(A,a) \arrow[r, "m"] \arrow[d, "f"'] \arrow[dr, phantom, "(PO)"] & (B,b) \arrow[d, "g"] \\
(C,c) \arrow[r, "n"'] & (D,d)
\end{tikzcd}$$

Proof. Given f, m with $m \in \mathcal{M}_{strict}$ we construct D as pushout object in **Sets**, which means $D = (C \dot{\cup} B)\backslash m(A)$ with inclusion $n : C \to D$, and $g : B \to D : b_1 \in B\backslash m(A) \mapsto b_1, m(a_1) \mapsto f(a_1)$. For $d_1 \in D$, d is defined by

(1) $d_1 = b_1 \in B\backslash m(A)$: $d(b_1) = b(b_1)$,

(2) $d_1 = c_1 \in C$: $d(c_1) = c(c_1)$.

Obviously, $d : D \to \mathbb{N}$ is well-defined.

First we shall show that g, n are **Markings**-morphisms and n is strict.

(i) $\forall b_1 \in B$ we have:
   1. $b_1 \in B\backslash m(A)$ and $b(b_1) \overset{(1)}{=} d(b_1) = d(g(b_1))$ or
   2. $\exists a_1 \in A$ with $b_1 = m(a_1)$ and $b(b_1) = b(m(a_1)) \overset{m \text{ strict}}{=} a(a_1) \leq c(f(a_1)) \overset{(2)}{=} d(f(a_1)) = d(g(m(a_1))) = d(g(b_1))$.

   This means $g \in$ **Markings**.

(ii) $\forall c_1 \in C$ we have:
   1. $c(c_1) \overset{(2)}{=} d(c_1) = d(n(c_1))$.

   This means $n \in$ **Markings** and n is strict.

It remains to show the pushout property. Given **Markings**-morphisms $h : (C,c) \to (E,e)$, $k : (B,b) \to (E,e)$ with $h \circ f = k \circ m$, we have a unique induced morphism x in **Sets** with $x \circ n = h$ and $x \circ g = k$. We shall show that $x \in$ **Markings**, i.e. $\forall d_1 \in D : d(d_1) \leq e(x(d_1))$.

$$\begin{tikzcd}
(A,a) \arrow[r, "m"] \arrow[d, "f"'] \arrow[dr, phantom, "(PO)"] & (B,b) \arrow[d, "g"'] \arrow[ddr, "k"] & \\
(C,c) \arrow[r, "n"] \arrow[drr, "h"'] & (D,d) \arrow[dr, "x"] & \\
 & & (E,e)
\end{tikzcd}$$

(i) For $d_1 = b_1 \in B\backslash m(A)$ we have $d(b_1) \overset{(1)}{=} b(b_1) \leq e(k(b_1)) = e(x(g(b_1)) = e(x(b_1))$.

(ii) For $d_1 = c_1 \in C$ we have $d(c_1) \overset{(2)}{=} c(c_1) \leq e(h(c_1)) = e(x(n(c_1))) = e(x(c_1))$.

$\square$

As next property, we shall show that pullbacks along $\mathcal{M}_{strict}$-morphisms exist and preserve $\mathcal{M}_{strict}$-morphisms.

Lemma 4.5 *In* **Markings***, pullbacks along $\mathcal{M}_{strict}$-morphisms exist and preserve $\mathcal{M}_{strict}$, i.e. given morphisms g and n with n strict, then the pullback (PB) exists and m is also a strict morphism.*

$$\begin{tikzcd}
(A,a) \arrow[r, "m"] \arrow[d, "f"'] \arrow[dr, phantom, "(PB)"] & (B,b) \arrow[d, "g"] \\
(C,c) \arrow[r, "n"'] & (D,d)
\end{tikzcd}$$

Proof. Given g, n with $n \in \mathcal{M}_{strict}$ we construct A as pullback object in **Sets**, which means $A = g^{-1}(n(C))$ with inclusion $m : A \to B$ and $f : A \to C : a \mapsto n^{-1}(g(a))$. For all $a_1 \in A$, a is defined by

$$(*) \quad a(a_1) = b(m(a_1)).$$

Obviously, a is a well-defined marking. f is a well-defined function since n is injective. We have to show that f, m are **Markings**-morphisms and m is strict.

(i) $\forall a_1 \in A$ we have: $a(a_1) \overset{(*)}{=} b(m(a_1)) \leq d(g(m(a_1)) = d(n(f(a_1)) \overset{n \text{ strict}}{=} c(f(a_1))$.
This means $f \in$ **Markings**.

(ii) $\forall a_1 \in A$ we have: $a(a_1) \overset{(*)}{=} b(m(a_1))$.
This means $m \in$ **Markings** and m is strict.

It remains to show the pullback property. Given **Markings**-morphisms $h : (E,e) \to (C,c)$, $k : (E,e) \to (B,b)$ with $n \circ h = g \circ k$, we have a unique induced morphism x in **Sets** with $f \circ x = h$ and $m \circ x = k$. We shall show that $x \in$ **Markings**, i.e. $\forall e_1 \in E : e(e_1) \leq a(x(e_1))$.

$$\begin{tikzcd}
(E,e) \arrow[dr, "x"] \arrow[drr, "k"] \arrow[ddr, "h"'] & & \\
& (A,a) \arrow[r, "m"'] \arrow[d, "f"] \arrow[dr, phantom, "(PB)"] & (B,b) \arrow[d, "g"] \\
& (C,c) \arrow[r, "n"'] & (D,d)
\end{tikzcd}$$

For $e_1 \in E$ we have $e(e_1) \leq b(k(e_1)) = b(m(x(e_1))) \overset{m \text{ strict}}{=} a(x(e_1))$. □

It remains to show the weak VK property for **Markings**. We know that $(\mathbf{Sets}, \mathcal{M})$ is a weak adhesive HLR category for the class $\mathcal{M}$ of injective morphisms [2], hence pushouts in **Sets** along injective morphisms are van Kampen squares. But we have to give an explicit proof for the markings, because a square (1) in **Markings** with $m, n \in \mathcal{M}_{strict}$, which is a pushout in **Sets**, is not necessarily pushout in **Markings**, since we may have $d(g(b_1)) > b(b_1)$ for some $b_1 \in B \backslash m(A)$.

$$\begin{tikzcd}
(A,a) \arrow[r, "m"] \arrow[d, "f"'] \arrow[dr, phantom, "(1)"] & (B,b) \arrow[d, "g"] \\
(C,c) \arrow[r, "n"'] & (D,d)
\end{tikzcd}$$

Lemma 4.6 *In* **Markings***, pushouts along* $\mathcal{M}_{strict}$*-morphisms are weak van Kampen squares.*

Proof. Given the following commutative cube (2) with $m \in \mathcal{M}_{strict}$ and ($f \in \mathcal{M}_{strict}$ or $t, u, v \in \mathcal{M}_{strict}$), where the bottom face is a pushout and the back faces are pullbacks, we have to show that the top face is a pushout if and only if the front faces are pullbacks.

$$\begin{tikzcd}
 & & (A',a') \arrow[dll, "f'"'] \arrow[dr, "m'"] \arrow[dd, "s"'] & \\
(C',c') \arrow[dr, "n'"] \arrow[dd, "u"] & & & (B',b') \arrow[dll, "g'"'] \arrow[dd, "t"] \\
 & (D',d') \arrow[dd, "v"] & (A,a) \arrow[dll, "f"'] \arrow[dr, "m"] & \\
(C,c) \arrow[dr, "n"'] & & & (B,b) \arrow[dll, "g"] \\
(2) & (D,d) & &
\end{tikzcd}$$

"$\Rightarrow$" If the top face is a pushout then the front faces are pullbacks in **Sets**, since all squares are pushouts or pullbacks in **Sets**, respectively, where the weak VK property holds. For a pullback (1) with $m, n \in \mathcal{M}_{strict}$, the function a of A is completely determined by the fact that $m \in \mathcal{M}_{strict}$ as shown in the proof of Lemma 4.5. Hence a diagram (1) in **Markings** with $m, n \in \mathcal{M}_{strict}$ is a pullback in **Markings** if and only if it is a pullback in **Sets**. This means, the front faces are also pullbacks in **Markings**.

"$\Leftarrow$" If the front faces are pullbacks we know that the top face is a pushout in **Sets**. To show that it is also a pushout in **Markings** we have to verify the conditions (1) and (2) from the construction in Lemma 4.4.

(1) For $b_1' \in B' \backslash m'(A')$ we have to show that $d'(g'(b_1')) = b'(b_1')$.

If f is strict then also g and g' are strict, since the bottom face is a pushout and the right front face is a pullback, and $\mathcal{M}_{strict}$ is preserved by both pushouts and pullbacks. This means that $b'(b_1') = d'(g'(b_1'))$.

Otherwise t and v are strict. Since the right back face is a pullback and $b_1' \in B' \backslash m'(A')$ we have $t(b_1') \in B \backslash m(A)$. With the bottom face being a pushout we have by (1) in Lemma 4.4

$$(*)\ d(g(t(b_1'))) \overset{(1)}{=} b(t(b_1')).$$

It follows that $d'(g'(b_1')) \overset{v \text{ strict}}{=} d(v(g'(b_1'))) = d(g(t(b_1'))) \overset{(*)}{=} b(t(b_1')) \overset{t \text{ strict}}{=} b'(b_1')$.

(2) For $c'_1 \in C'$ we have to show that $d'(n'(c'_1)) = c'(c'_1)$.

With m being strict also n and n' are strict, since the bottom face is a pushout and the left front face is a pullback, and $\mathcal{M}_{strict}$ is preserved by both pushouts and pullbacks. This means that $c'(c'_1) = d'(n'(c'_1))$.

□

Theorem 4.7 *The category* $(\mathbf{Markings}, \mathcal{M}_{strict})$ *is a weak adhesive HLR category.*

Proof. By Lemma 4.3, the morphism class $\mathcal{M}_{strict}$ has the required properties. Moreover, we have pushouts and pullbacks along $\mathcal{M}_{strict}$-morphisms in **Markings**, as shown in Lemma 4.4 and Lemma 4.5, respectively. By Lemma 4.6, pushouts along strict morphisms are weak van Kampen squares. Hence all properties of weak adhesive HLR categories are fulfilled and $(\mathbf{Markings}, \mathcal{M}_{strict})$ is a weak adhesive HLR category. □

Remark 4.8 If we consider the full subcategory **B** of **Markings** consisting of objects (S, s) with $s : S \to \{0, 1\}$ only, this category can be shown to be a weak adhesive HLR category as well. With this category, we are able to describe the markings of elementary nets, where on each place at most one token is allowed.

5 From Nets to Systems

In this section we combine nets with markings and show that under certain conditions the category of the corresponding systems is also a weak adhesive HLR category. The term *net* means any variant of Petri nets, for example place/transition nets, AHL nets or generalized AHL nets.

The general idea is to define for a net N a marking set $M(N)$ dependent on N, where the actual marking is a function $m : M(N) \to \mathbb{N}$. For place/transition nets this marking set is the set P of places, for AHL nets and generalized AHL nets this marking set is the set $(A \otimes P)$. Then the category of the corresponding systems can be seen as a subcategory of a comma category of nets and markings, where the marking set is compatible with the net.

Definition 5.1 (System) Given a category **Nets** of nets, a system $S = (N, m)$ is given by a net $N \in \mathbf{Nets}$ and a function $m : M(N) \to \mathbb{N}$, where $M : \mathbf{Nets} \to \mathbf{Sets}$ is a functor assigning a marking set to each net N.

For systems $S = (N, m)$ and $S' = (N', m')$, a system morphism $f_S : S \to S'$ is a net morphism $f_N : N \to N'$ such that $M(f_N) : (M(N), m) \to (M(N'), m')$ is a **Markings**-morphism.

Systems and system morphisms form the category **Systems**.

Theorem 5.2 *Given a weak adhesive HLR category* $(\mathbf{Nets}, \mathcal{M}')$ *of nets with a marking set functor* $M : \mathbf{Nets} \to \mathbf{Sets}$ *that preserves pushouts and pullbacks along* $\mathcal{M}'$*-morphisms, then the category* $(\mathbf{Systems}, \mathcal{M})$ *of systems over these nets is a weak adhesive HLR category, where* $\mathcal{M}$ *is the class of all morphisms* $f_S = (f_N)$ *with* $f_N \in \mathcal{M}'$ *and* $M(f_N) \in \mathcal{M}_{strict}$.

Proof. First we define the category $\mathbf{C} = ComCat(M, V, \{1\})$ with $V : \mathbf{Markings} \to \mathbf{Sets}, V(T, t) = T, V(f) = f$. We can apply Thm. 2.3.(v) using that

M preserves pushouts along $\mathcal{M}'$ and V preserves pullbacks along $\mathcal{M}_{strict}$, which follows from the construction in the proof of Lemma 4.5. It follows that $(\mathbf{C}, \mathcal{M}_C)$ with $\mathcal{M}_C = (\mathcal{M}' \times \mathcal{M}_{strict})|_{\mathbf{C}}$ is a weak adhesive HLR category.

Now we only consider objects $(N, (T, t), op^1) \in \mathbf{C}$ where $op^1 : M(N) \to T$ is an identity, i.e. $M(N) = T$. This restriction leads to the full subcategory $\mathbf{D}$ of $\mathbf{C}$. By construction, the category $\mathbf{D}$ is isomorphic to the category **Systems**:

- For an object $D = (N, (T, t), op^1) \in \mathbf{D}$ we have $op^1 : M(N) \to T$ is an identity, i.e. $D = (N, (M(N), t : M(N) \to \mathbb{N}), id_{M(N)})$, which is a one-to-one correspondence to the system $(N, t) \in \mathbf{Systems}$.
- For a morphism $f = (f_N, f_M) : D \to D'$ with $D = (N, (T, t), op^1)$ and $D' = (N', (T', t'), op^{1'})$ we have $D = (N, (M(N), t : M(N) \to \mathbb{N}), id_{M(N)})$ and $D' = (N', (M(N'), t' : M(N') \to \mathbb{N}), id_{M(N')})$, and by the definition of morphisms in a comma category $id_{M(N')} \circ M(f_N) = V(f_M) \circ id_{M(N)}$. This means that $M(f_N) = V(f_M)$, which corresponds to the morphism $f_S = (f_N) \in \mathbf{Systems}$, where $M(f_N)$ is a **Markings**-morphism.

To apply Thm. 2.3.(i), we have to show that $\mathbf{D}$ has pushouts and pullbacks along $\mathcal{M}_D$-morphisms with $\mathcal{M}_D = \mathcal{M}_C|_{\mathbf{D}}$ that are preserved by the inclusion functor. Given objects $(N_i, (M(N_i), m_i), op_i^1 = id_{M(N_i)})$ for $i = 0, 1, 2$ and morphisms $f_S = (f_N, f_M) : (N_0, (M(N_0), m_0), op_0^1) \to (N_1, (M(N_1), m_1), op_1^1)$ and $g_S = (g_N, g_M) : (N_0, (M(N_0), m_0), op_0^1) \to (N_2, (M(N_2), m_2), op_2^1)$ with $f_S \in \mathcal{M}_D$ we can construct the pushout (1) of f_N, g_N in **Nets** with $f_N \in \mathcal{M}'$. Since M preserves pushouts along $\mathcal{M}'$-morphisms, (2) is a pushout in **Sets**. By assumption, we have $M(f_N) \in \mathcal{M}_{strict}$. Now we can use the construction in the proof of Lemma 4.4 to construct a marking $m_3 : M(N_3) \to \mathbb{N}$ leading to the pushout (3) in **Markings**. By the construction of pushouts in comma categories, $(N_3, (M(N_3), m_3), op_3^1 = id_{M(N_3)})$ is a pushout in $\mathbf{C}$ and $\mathbf{D}$.

$$\begin{tikzcd}
N_0 \arrow[r, "f_N"] \arrow[d, "g_N"'] \arrow[dr, phantom, "(1)"] & N_1 \arrow[d, "g'_N"] & M(N_0) \arrow[r, "M(f_N)"] \arrow[d, "M(g_N)"'] \arrow[dr, phantom, "(2)"] & M(N_1) \arrow[d, "M(g'_N)"] & (M(N_0), m_0) \arrow[r, "M(f_N)"] \arrow[d, "M(g_N)"'] \arrow[dr, phantom, "(3)"] & (M(N_1), m_1) \arrow[d, "M(g'_N)"] \\
N_2 \arrow[r, "f'_N"'] & N_3 & M(N_2) \arrow[r, "M(f'_N)"'] & M(N_3) & (M(N_2), m_2) \arrow[r, "M(f'_N)"'] & (M(N_3), m_3)
\end{tikzcd}$$

Analogously, this can be done for pullbacks using the fact that M preserves pullbacks along $\mathcal{M}'$-morphisms and the construction of pullbacks in **Markings**.

This means that we can apply Thm. 2.3.(i) and $(\mathbf{Systems}, \mathcal{M}) \stackrel{\sim}{=} (\mathbf{D}, \mathcal{M}_D)$ is a weak adhesive HLR category. $\square$

As stated in Cor. 3.6 and already shown in [10], the category **PTSystems** with the class $\mathcal{M}$ of all strict morphisms is a weak adhesive HLR category. This follows directly from Thm. 5.2, since $(\mathbf{PTNets}, \mathcal{M}')$ is a weak adhesive HLR category and $M : \mathbf{PTNets} \to \mathbf{Sets}$ with $M(P, T, pre, post) = P$ preserves pushouts and pullbacks along $\mathcal{M}'$-morphisms because pushouts and pullbacks in **PTNets** along injective morphisms are constructed componentwise in **Sets**. Similarly, using Rem. 4.8, we obtain that elementary systems with the class $\mathcal{M}$ of all strict morphisms form a weak adhesive HLR category.

Now we can apply Thm. 5.2 to AHL systems and show that AHL systems, with a suitable choice of algebras and $\mathcal{M}$-morphisms, are a weak adhesive HLR category.

Definition 5.3 Given an algebraic specification SP, an *AHL system* $AS = (AN, m)$ is given by an AHL net $AN = (P, T, pre, post, cond, type, A)$ over SP with $A \in \mathbf{A(SP)}$, where $\mathbf{A(SP)}$ is a subcategory of $\mathbf{Algs(SP)}$, and a marking $m : (A \otimes P) \to \mathbb{N}$.

An *AHL system morphism* $f_{AS} : AS \to AS'$ is given by an AHL net morphism $f_{AN} = (f_{AC}, f_A) : AN \to AN'$ with $f_{AC} = (f_P, f_T)$ and $f_A \in \mathbf{A(SP)}$ that is marking-preserving, i.e. $\forall (a,p) \in A \otimes P : m(a,p) \leq m'(f_A(a), f_P(p))$.

AHL systems and AHL system morphisms form the category $\mathbf{AHLSystems(SP)}$.

Theorem 5.4 *If $(\mathbf{AHLNets(SP)}, \mathcal{M}')$ is a weak adhesive HLR category and the functor $M : \mathbf{AHLNets(SP)} \to \mathbf{Sets}$, defined by $M(P, T, pre, post, cond, type, A) = A \otimes P$ and $M(f_N) = f_A \otimes f_P$ for $f_{AN} = (f_{AC}, f_A)$ and $f_{AC} = (f_P, f_T)$, preserves pushouts and pullbacks along $\mathcal{M}'$-morphisms, then the category $(\mathbf{AHLSystems(SP)}, \mathcal{M})$ is a weak adhesive HLR category, where $\mathcal{M}$ is the class of all strict morphisms, i.e. $f_{AS} = (f_{AC}, f_A) : AS \to AS' \in \mathcal{M}$ if $f_A \in \mathcal{M}_1$, $f_{AC} = (f_P, f_T)$ is injective and f_{AS} is marking-strict, i.e. $\forall (a,p) \in A \otimes P : m(a,p) = m'(f_A(a), f_P(p))$.*

Proof. By Cor. 3.10, the category $(\mathbf{AHLNets(SP)}, \mathcal{M}')$ with a suitable choice of algebras is a weak adhesive HLR category. Then we can apply Thm. 5.2 to obtain the result that $(\mathbf{AHLSystems(SP)}, \mathcal{M})$ is a weak adhesive HLR category. □

Unfortunately, the condition that M has to preserve pushouts and pullbacks along $\mathcal{M}'$-morphisms is very strict and up to now only two suitable choices for the category $(\mathbf{AHLNets(SP)}, \mathcal{M}')$ are known:

(i) The category $(\mathbf{AHLNets(SP)}, \mathcal{M}_{iso})$ with the class $\mathcal{M}_{iso}$ of isomorphisms.

Given the pushout square (1) along an isomorphism f_{AN} in $\mathbf{AHLNets(SP)}$, also f'_{AN} is an isomorphism. It follows that $(f_A \otimes f_P)$ and $(f'_A \otimes f'_P)$ are isomorphisms and (2) is a pushout in $\mathbf{Sets}$. This can be done analogously for pullbacks, therefore M preserves pushouts and pullbacks along $\mathcal{M}_{iso}$-morphisms.

$$\begin{tikzcd} AN_0 \arrow[r, "f_{AN}"] \arrow[d, "g_{AN}"'] \arrow[dr, phantom, "(1)"] & AN_1 \arrow[d, "g'_{AN}"] \\ AN_2 \arrow[r, "f'_{AN}"'] & AN_3 \end{tikzcd} \qquad \begin{tikzcd} (A_0 \otimes P_0) \arrow[r, "(f_A \otimes f_P)"] \arrow[d, "(g_A \otimes g_P)"'] \arrow[dr, phantom, "(2)"] & (A_1 \otimes P_1) \arrow[d, "(g'_A \otimes g'_P)"] \\ (A_2 \otimes P_2) \arrow[r, "(f'_A \otimes f'_P)"'] & (A_3 \otimes P_3) \end{tikzcd}$$

Note, that not only the algebra homomorphism but also the net morphism has to be an isomorphism. This means that this morphism class is not useful for adhesive HLR systems, since only isomorphic productions would be allowed.

(ii) The category $(\mathbf{AHLNets(SP, A_{fin})}, \mathcal{M}_{inj})$ of algebraic high-level nets with a finite, fixed algebra A, with the class $\mathcal{M}_{inj}$ of injective morphisms with identities on the algebra part.

Given the pushout square (1), we have to show that also (2) is a pushout.

$$\begin{tikzcd} AN_0 \arrow[r, "f_{AN}"] \arrow[d, "g_{AN}"'] \arrow[dr, phantom, "(1)"] & AN_1 \arrow[d, "g'_{AN}"] \\ AN_2 \arrow[r, "f'_{AN}"'] & AN_3 \end{tikzcd} \qquad \begin{tikzcd} (A \otimes P_0) \arrow[r, "(id_A \otimes f_P)"] \arrow[d, "(id_A \otimes g_P)"'] \arrow[dr, phantom, "(2)"] & (A \otimes P_1) \arrow[d, "(id_A \otimes g'_P)"] \\ (A \otimes P_2) \arrow[r, "(id_A \otimes f'_P)"'] & (A \otimes P_3) \end{tikzcd}$$

For $s \in S$, define $P_s = \{p \in P \mid type(p) = s\}$ and $f_s = f|_{P_s}$ for a morphism $f : P \to P'$. Then we have for an AHL net $AN = (P, T, pre, post, cond, type, A)$ that $M(AN) = (A \otimes P) = \uplus_{s \in S}(A_s \otimes P_s) = \uplus_{s \in S}(\uplus_{a \in A_s}(\{a\} \times P_s))$.

Using the weak van Kampen property and the fact that (3) is a pushout, also (4) is a pushout. For $a \in A_s$ the square $(4a)$ is isomorphic to (4) and thus also a pushout. Since coproducts of pushout squares are again pushouts, also $(2) = \uplus_{s \in S}(\uplus_{a \in A_s}(4a))$ is a pushout.

$$\begin{tikzcd} P_0 \arrow[r, "f_P"] \arrow[d, "g_P"'] \arrow[dr, phantom, "(3)"] & P_1 \arrow[d, "g'_P"] \\ P_2 \arrow[r, "f'_P"'] & P_3 \end{tikzcd} \quad \begin{tikzcd} P_{0,s} \arrow[r, "f_{P,s}"] \arrow[d, "g_{P,s}"'] \arrow[dr, phantom, "(4)"] & P_{1,s} \arrow[d, "g'_{P,s}"] \\ P_{2,s} \arrow[r, "f'_{P,s}"'] & P_{3,s} \end{tikzcd} \quad \begin{tikzcd} \{a\} \times P_{0,s} \arrow[r, "id \times f_{AN}"] \arrow[d, "id \times g_{AN}"'] \arrow[dr, phantom, "(4a)"] & \{a\} \times P_{1,s} \arrow[d, "id \times g'_{AN}"] \\ \{a\} \times P_{2,s} \arrow[r, "id \times f'_{AN}"'] & \{a\} \times P_{3,s} \end{tikzcd}$$

Analogously this can be done for pullbacks, since coproducts of pullback squares are again pullbacks in **Sets**.

Even in the case of algebras without operations and the class $\mathcal{M}'$ of injective morphisms with isomorphic data part, the marking set functor M does not preserve pushouts along $\mathcal{M}'$-morphisms, as shown in Fig. 1 with $f \in \mathcal{M}'$, where all places are typed by *nat*. The square (1) is a pushout in **AHLNets**(**SP**), but the square (2) as the result of applying M to (1) is no pushout in **Sets**. This is due to the fact that the product of two pushouts in **Sets** in general does not yield a pushout.

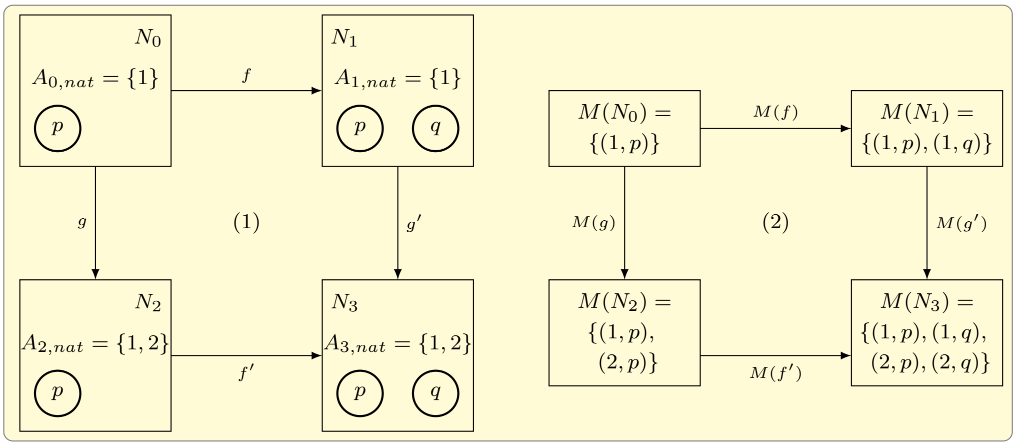


Fig. 1. Pushout along $\mathcal{M}'$-morphism f that is not preserved by the marking set functor M

Analogously, we can show that generalized AHL systems form a weak adhesive HLR category, if the marking set functor M preserves pushouts and pullbacks along $\mathcal{M}'$-morphisms.

Definition 5.5 A *generalized AHL system* $GS = (GN, m)$ is given by a generalized AHL net $GN = (SP, P, T, pre, post, cond, type, A)$ with $A \in \mathbf{A}$, where $\mathbf{A}$ is a subcategory of **Algs**, and a marking $m : (A \otimes P) \to \mathbb{N}$.

A *generalized AHL system morphism* $f_{GS} : GS \to GS'$ is given by a generalized AHL net morphism $f_{GN} = (f_{GC}, f_{GA}) : GN \to GN'$ with $f_{GC} = (f_P, f_T)$ and $f_{GA} \in \mathbf{A}$ that is marking-preserving, i.e. $\forall (a,p) \in A \otimes P : m(a,p) \leq m'(f_A(a), f_P(p))$.

Generalized AHL systems and generalized AHL system morphisms form the category **AHLSystems**.

Theorem 5.6 *If* $(\mathbf{AHLNets}, \mathcal{M}')$ *is a weak adhesive HLR category and the functor* $M : \mathbf{AHLNets} \to \mathbf{Sets}$ *with* $M(SP, P, T, pre, post, cond, type, A) = A \otimes P$ *and* $M(f_{GN}, f_{GA}) = f_{GA} \otimes f_P$ *for* $f_{GN} = (f_{GC}, f_{GA})$ *and* $f_{GC} = (f_P, f_T)$ *preserves pushouts and pullbacks along* $\mathcal{M}'$*-morphisms, then the category* $(\mathbf{AHLSystems}, \mathcal{M})$ *is a weak adhesive HLR category, where* $\mathcal{M}$ *is the class of all strict morphisms, i.e.* $f_{GS} = (f_{GC}, f_{GA}) : GS \to GS' \in \mathcal{M}$ *if* $f_{GA} \in \mathcal{M}_1$*,* $f_{GC} = (f_P, f_T)$ *is strict injective and* f_{GS} *is marking-strict, i.e.* $\forall (a,p) \in A \otimes P : m(a,p) = m'(f_A(a), f_P(p))$.

Proof. By Cor. 3.14, $(\mathbf{AHLNets}, \mathcal{M}')$ with a suitable choice of algebras is a weak adhesive HLR category. Then we can apply Thm. 5.2 to obtain the result that $(\mathbf{AHLSystems}, \mathcal{M})$ is a weak adhesive HLR category. □

Analogously to the case of $\mathbf{AHLNets(SP)}$, the conditions for M are very rerestrictive, so only two suitable choices for the category $(\mathbf{AHLNets}, \mathcal{M}')$ are known up to now:

(i) The category $(\mathbf{AHLNets}, \mathcal{M}_{iso})$ with the class $\mathcal{M}_{iso}$ of isomorphisms, which is, analogously to the case $(\mathbf{AHLNets(SP)}, \mathcal{M}_{iso})$, not useful for adhesive HLR systems.

(ii) The category $(\mathbf{AHLNets_{iso}}, \mathcal{M}_{sinj})$ of algebraic high-level nets with morphisms that are isomorphisms on the algebra part, with the class $\mathcal{M}_{inj}$ of strict injective morphisms.

6 Example for a Reconfigurable AHL System

In this section we present an example for a reconfigurable AHL system, i.e. an AHL system that can be transformed by rule applications.

Our goal is to model a small library system, where users may enter the reading hall if they have an access card and do not carry a bag. In the reading hall, they may read books if they have a library card. For leaving the reading hall they have to give back all books.

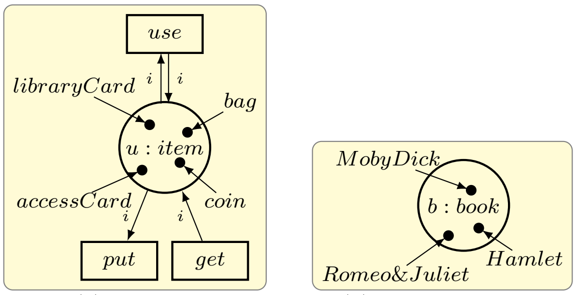
The specification SP with empty set of equations and the algebra A are given in Fig. 2. We define various items that users may carry and some books in the reading hall.

A user outside the reading hall is modeled by the net in Fig. 3(a), where the tokens at the place u are the items she carries around, which can be used by firing the transition *use*. A user can also interact with its environment by receiving some items firing the transition *get* or by delivering some items firing the transition *put*.

	SP	A
sorts:	$item$ $book$	$A_{item} = \{accessCard, libraryCard, bag, coin\}$ $A_{book} = \{Hamlet, MobyDick, Romeo\&Juliet\}$
opns:	$LC :\to item$	$LC_A = libraryCard \in A_{item}$
vars:	$i : item$ $b : book$	

Fig. 2. Specification and algebra for the library example

The reading hall is modeled by a place b as shown in Fig. 3(b) where the tokens at b are all the available books.



(a) User net (b) Reading hall net

Fig. 3. Nets for the different parts of the library system

For a concrete example we construct the disjoint unions of the reading hall net and the user nets (for two different users), which is shown in Fig. 4.

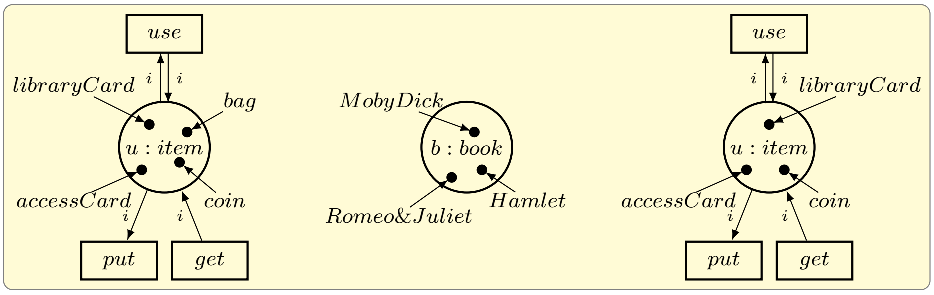


Fig. 4. Start net for the library system with two users

The movement of the users is modeled by productions, where we use, in addition to Def. 2.4, productions with negative application conditions. A negative application condition forbids a certain structure in the net extending the match. Formally, given a production $p = (L \overset{l}{\leftarrow} K \overset{r}{\rightarrow} R)$ it is an object N and a morphism $n : L \to N$. In this case, p can be applied to G via a match $m : L \to G$ if $G \overset{p,m}{\Longrightarrow} G'$ is a direct transformation and we do not find a morphism $q : N \to G$ such that $q \circ n = m$.

$$\begin{tikzcd}
N \arrow[dr, "q"', "/" marking] & L \arrow[l, "n"'] \arrow[d, "m"'] \arrow[dr, phantom, "(1)"] & K \arrow[l, "l"'] \arrow[r, "r"] \arrow[d, "k"] \arrow[dr, phantom, "(2)"] & R \arrow[d, "n"] \\
 & G & D \arrow[l, "f"] \arrow[r, "g"'] & H
\end{tikzcd}$$

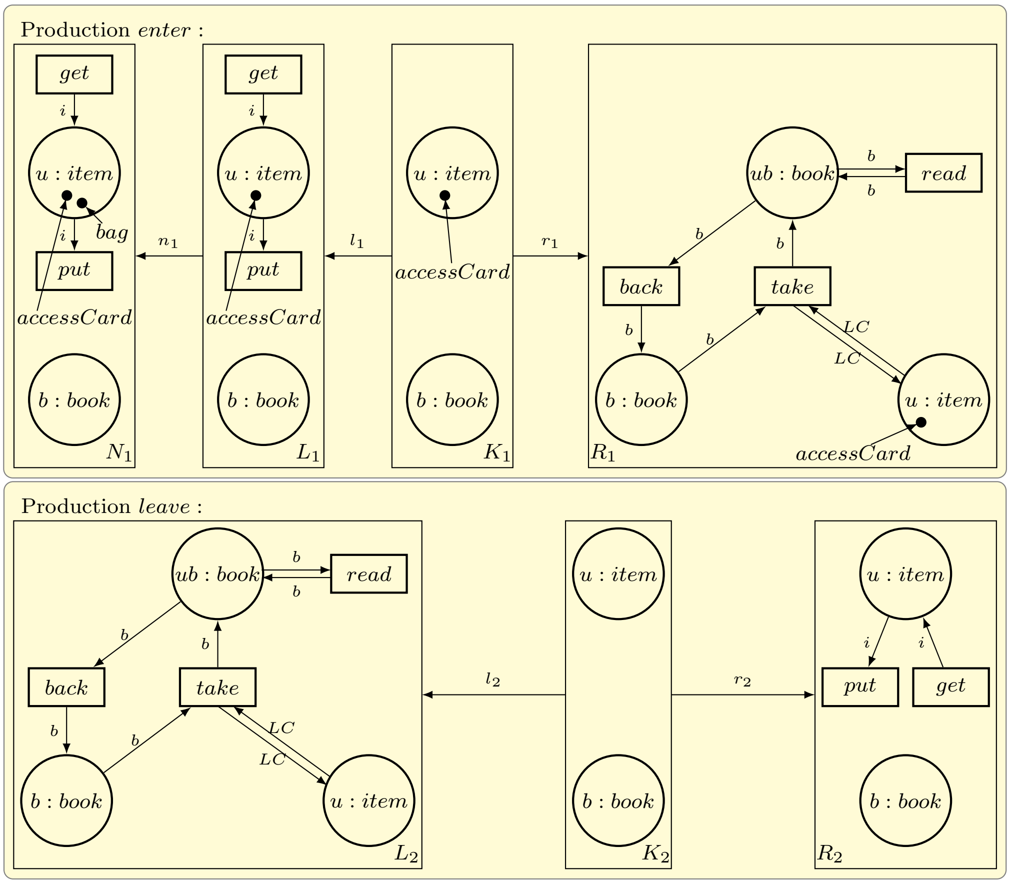


Fig. 5. Productions for entering and leaving the reading hall

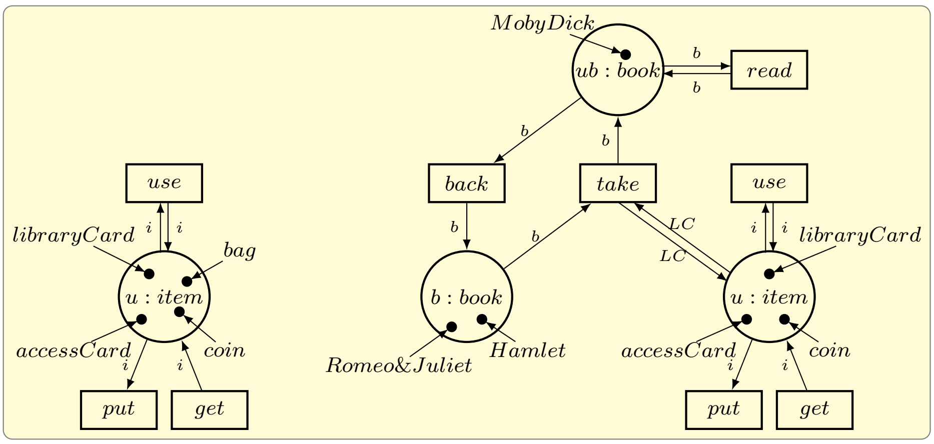


Fig. 6. Library system after the application of the production *enter* and the firing of *take*

The productions *enter* and *leave* in Fig. 5 handle the use of the reading hall, where all morphisms are inclusions given by the names of the elements. With *enter*, a user may enter the reading hall. In this case, she needs to carry an access card, modeled by the token *accessCard* on u, and is not allowed to carry a bag, which is modeled by a negative application condition. After the application of the production, if she has a library card she is able to take books by firing the new transition *take* and to read them by firing the transition *read*. She can no longer interact with her environment, so we delete the transitions *put* and *get*. If she has given back all books, she may leave the reading hall by applying the inverse production *leave*. This restriction does not have to be modeled by a negative application condition, because due to the pushout construction the production is only applicable if there is no token at the place *ub*. In Fig. 6, the application of the production *enter* to the right user in Fig. 4 followed by a firing step *take* is shown. Now this user may read Moby Dick. The production *enter* cannot be applied to the left user, because she does have a bag. But she might use the transition *put* to leave the bag and, afterwards, she may go to the reading hall.

7 Conclusion and Future Work

In this paper we have introduced the category of markings for Petri nets, which gives a general construction for the marking of a net. Using this category, we have extended different variants of Petri nets to systems, i.e. nets with markings. In particular this works for place/transition systems, AHL systems and generalized AHL systems and we have shown that these systems are weak adhesive HLR categories for a suitable choice of markings and $\mathcal{M}$-morphisms. This means that we can apply the theory for graph transformations developed in [2] also to different kinds of net transformations based on AHL schemas, nets, and systems.

There are many different notions of Petri nets, AHL nets etc., which have been shown to be weak adhesive HLR categories. In Fig. 7, an overview of the available results for schemas, nets and systems is given.

specification/algebra	schemas	nets	systems
none, black token	—	$\mathcal{M}_{injective}$	$\mathcal{M}_{strict}$
elementary nets	—	$\mathcal{M}_{injective}$	$\mathcal{M}_{injective}$
fixed specification, fixed algebra	$\mathcal{M}_{injective}$	$\mathcal{M}_{injective}$	$\mathcal{M}_{strict}$
fixed specification, general algebra	$\mathcal{M}_{injective}$	$\mathcal{M}_{injective}$, $(\mathbf{A}(\mathbf{SP}), \mathcal{M})$ wAHLR	$\mathcal{M}_{injective}$, $(\mathbf{A}(\mathbf{SP}), \mathcal{M}')$ wAHLR, M preserves PO, PB
general specification, general algebra	$\mathcal{M}_{strict\ injective}$	$\mathcal{M}_{strict\ injective}$, $(\mathbf{A}, \mathcal{M}')$ wAHLR	$\mathcal{M}_{strict\ injective}$, $(\mathbf{A}, \mathcal{M}')$ wAHLR, M preserves PO, PB

Fig. 7. Requirements for schemas, nets and systems to be weak adhesive HLR categories

At the moment, the available data structure underlying the AHL nets is restricted to a few, but still interesting cases. More work is needed in the area of algebras, where the categories $\mathbf{Algs}(\mathbf{SP})$ of algebras over a certain specification SP and $\mathbf{Algs}$ of generalized algebras and homomorphisms should be verified to be weak adhesive HLR categories, likely under some restrictions on the specification or $\mathcal{M}$-morphisms. The category $\mathbf{Algs}$ is equivalent to a Grothendieck category (see [12]) indexed over the category $\mathbf{Specs}$. Grothendieck categories have general pushouts and pullbacks, if so have the underlying categories, and they have free constructions, but they have not been shown to be weak adhesive HLR categories. A step towards this has been made in [5], where also some restrictions to the morphism class $\mathcal{M}$ are discussed which could lead to a suitable weak adhesive HLR category.

In addition, it should be analyzed if there are other categorical constructions leading to Petri net systems that can be shown to be weak adhesive HLR categories, where the very strict conditions on the marking set functor M can be weakened.

References

[1] Ehrig, H., *Introduction to the Algebraic Theory of Graph Grammars (A Survey)*, in: V. Claus, H. Ehrig and G. Rozenberg, editors, *Graph Grammars and their Application to Computer Science and Biology*, LNCS **73** (1979), pp. 1–69.

[2] Ehrig, H., K. Ehrig, U. Prange and G. Taentzer, "Fundamentals of Algebraic Graph Transformation," EATCS Monographs, Springer, 2006.

[3] Ehrig, H., A. Habel, H. Kreowski and F. Parisi-Presicce, *Parallelism and Concurrency in High-Level Replacement Systems*, Mathematical Structures in Computer Science **1** (1991), pp. 361–404.

[4] Ehrig, H., A. Habel, J. Padberg and U. Prange, *Adhesive High-Level Replacement Systems: A New Categorical Framework for Graph Transformation*, Fundamenta Informaticae **74** (2006), pp. 1–29.

[5] Ehrig, H., F. Orejas and U. Prange, *Categorical Foundations of Distributed Graph Transformation*, in: A. Corradini, H. Ehrig, U. Montanari, L. Ribeiro and G. Rozenberg, editors, *Graph Transformations. Proceedings of ICGT 2006*, LNCS **4178** (2006), pp. 215–229.

[6] Hoffmann, K., H. Ehrig and T. Mossakowski, *High-Level Nets with Nets and Rules as Tokens*, in: G. Ciardo and P. Darondeau, editors, *Applications and Theory of Petri Nets. Proceedings of ATPN 2005*, LNCS **3536** (2005), pp. 268–288.

[7] Lack, S. and P. Sobociński, *Adhesive and Quasiadhesive Categories*, Theoretical Informatics and Applications **39** (2005), pp. 511–546.

[8] Padberg, J., H. Ehrig and L. Ribeiro, *Algebraic High-Level Net Transformation Systems*, Mathematical Structures in Computer Science **5** (1995), pp. 217–256.

[9] Prange, U., *Algebraic High-Level Nets as Weak Adhesive HLR Categories*, Electronic Communications of the EASST **2** (2007), to appear.

[10] Prange, U., H. Ehrig, K. Hoffmann and J. Padberg, *Towards Concurrency in Reconfigurable Place/Transition Systems*, 2007, submitted.

[11] Sobociński, P., "Deriving Process Congruences from Reaction Rules," Ph.D. thesis, BRICS (2004).

[12] Tarlecki, A., R. Burstall and J. Goguen, *Some Fundamental Algebraic Tools for the Semantics of Computation: Part 3: Indexed Categories*, Theoretical Computer Science **91** (1991), pp. 239–264.